Diffusion-based spatial priors for ima

Lee Michael Harrison

BSc(Hons), BM, MRCS(Ed)

Wellcome Trust Centre for Neuroimaging

Institute of Neurology

University College London

12 Queen Square

London WC1N 3BG, United Kingdom

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

of the

University College London

2008

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

I, Lee Michael Harrison, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

Abstract

We describe a Bayesian scheme to analyze images, which uses spatial priors encoded by a diffusion kernel, based on a weighted graph Laplacian. This provides a general framework to formulate a spatial model, whose parameters can be optimised. The standard practice using the software statistical parametric mapping (SPM) is to smooth imaging data using a fixed Gaussian kernel as a pre-processing step before applying a mass-univariate statistical model (*e.g.,* a general linear model) to provide images of parameter estimates (Friston et al., 2006). This entails the strong assumption that data are generated smoothly throughout the brain. An alternative is to include smoothness in a multivariate statistical model (Penny et al., 2005). The advantage of the latter is that each parameter field is smoothed automatically, according to a measure of uncertainty, given the data. Explicit spatial priors enable formal model comparison of different prior assumptions, *e.g.* that data are generated from a stationary (*i.e.* fixed throughout the brain) or non-stationary spatial process. We describe the motivation, background material and theory used to formulate diffusion-based spatial priors for fMRI data and apply it to three different datasets, which include standard and high-resolution data. We compare mass-univariate ordinary least squares estimates of smoothed data and three Bayesian models; spatially independent, stationary and non-stationary spatial models of non-smoothed data. The latter of which can be used to preserve boundaries between functionally selective regional responses of the brain, thereby increasing the spatial detail of inferences about cortical responses to experimental input.

Acknowledgements

I would like to thank my supervisor, Karl Friston, for his continual inspiration and generosity throughout my time at the Wellcome Trust Centre for Neuroimaging. I am very grateful to Richard Frackowiak for introducing me to Karl, which marked my point of departure from a clinical to scientific career. Will Penny has been like a second supervisor in giving well timed suggestions and encouragement. I have enjoyed many amusing conversations with Jean Daunizeau during our cigarette breaks and thank my close friend Klaas Stephan for being just that and have thoroughly enjoyed our work together. I cannot imagine life in London without Rico and Dave, who have always been close by. Marcia has been brilliant at organizing trips abroad and generally making life easier. The Wellcome Trust has supported me financially through Karl's Programme Grant, to whom I am grateful.

I am indebted to my family. Even though my grand parents are not with us now, I am deeply grateful for their devotion to me as a child and young man. Aunty Barbara has always been an inspiration, providing me with timely glimpses of the wondrous variety out there in the world and a way to be within it. Lastly, I thank my parents, Mum, Chris and Dad, for their unconditional love that has and will always be my safety net.

Contents

List of figures

List of tables

Abbreviations

BOLD	-	blood oxygenation level dependence
CSF	-	cerebral spinal fluid
EGL	-	Euclidean graph-Laplacian
EPI	-	echo-planar image
fMRI	-	functional magnetic resonance imaging
FWHM	-	full width at half maximum
GGL	-	geodesic graph-Laplacian
GLM	-	general linear model
GMRF	-	Gaussian Markov random field
GPM	-	Gaussian process model
GPP	-	Gaussian process prior
GSP	-	global shrinkage prior
HRF	-	haemodynamic response function
i.i.d	-	independent and identically distributed
LBO	-	Laplace-Beltrami operator
MVN	-	matrix-variate normal (density)
OLS	-	ordinary least squares
pdf	-	probability density function
PPM	-	posterior probability map
RFT	-	random field theory
SNR	-	signal to noise ratio
SPM	-	statistical parametric mapping or a statistical parametric map
WGL	-	weighted graph-Laplacian

Notation

This list is not exhaustive. See also mathematical background in the Appendix.

Y	-	data matrix
X	-	design matrix, i.e. columns of explanatory variables
β	-	matrix containing GLM parameters (referred to as parameters)
α	-	set of hyper-parameters
$p(x, y)$	-	joint probability of random variables x and y
$\int_y p(x, y)$	-	marginal probability of x, where we have suppressed dy
$p(x \mid y)$	-	conditional probability of x, given y
$\langle f(x) \rangle_{p(x)}$	-	expectation of $f(x)$ under $p(x)$
V, E	-	vertex and edge sets (unless otherwise stated) of a graph
N_V, N_E	-	cardinality, *i.e.* number of elements, in the sets V and E
$i \sim j$	-	neighboring vertices
v_i, e_{ij}	-	elements of the sets V and E
w_{ij}, W	-	edge weight (between vertices i and j) and weight matrix
L	-	weighted graph-Laplacian (discrete Laplace operator), also known as the Laplacian matrix

$\exp(-L\tau)$	-	diffusion kernel of a (scaled) WGL, using the matrix exponential, where $\tau \in [0,\infty)$
$\dot{f} = \frac{\partial f}{\partial t}$	-	(partial) temporal derivative of a function $f(x,t)$
grad, ∇	-	gradient operator
div, $\nabla \cdot$	-	divergence operator
div(*grad*), Δ	-	Laplace operator
$\chi : M \to N$	-	map from manifold M to N, which have metrics G and H
G, H	-	induced and embedding space metric tensors
J	-	Jacobian matrix
$A \in \Re^{m \times n}$	-	A is a real matrix, size $m \times n$
I_n	-	identity matrix, size $n \times n$
I_{kl}	-	element of expected Fisher Information matrix
$\det(A)$	-	determinant of a matrix A
$tr(A)$	-	trace of a matrix A
A^T	-	transpose of a matrix A
$\otimes$	-	Kronecker delta product

Outline

The thesis is organized as follows. In Chapter 1, we introduce the problem of quantifying the spatial distribution of cortical responses to experimental input, given functional magnetic resonance imaging (fMRI) data using the current standard practice implemented in SPM. We provide an overview of the theoretical fundaments of our approach, which is (1) formulated in terms of probabilities, i.e. a Bayesian framework, (2) informed by methods used in image restoration and segmentation, e.g. edge preserving flows and (3) based on representing a parameter image as a random spatial process, i.e. a random field (RF). In Chapter 2 we review relevant material from graph theory, which is used throughout the thesis. This includes computing the weighted graph-Laplacian (WGL), its eigensystem and diffusion kernel, and describing how it can be used to partition a brain volume into sub-graphs. This latter step is pragmatic in that it reduces the computational load of our implementation by reducing the Laplacian matrix over a volume to a block-diagonal form. The edge weights play a crucial role as their dependence on a parameter image leads to an anisotropic (*i.e.* with a preferred direction) Laplacian matrix, without which it is isotropic. In Chapter 3 we describe the model in detail with emphasis on using diffusion (heat) kernels to represent covariances within a hierarchical observation model. We start with a two-level general linear model (GLM) with matrix-variate normal (MVN) density priors on GLM parameters. We focus on reducing the model to the specification of covariance components, in particular, the form of the covariance matrix and its hyper-parameters. We then look at the form of the spatial priors using diffusion kernels and relate our formulation to other schemes. An eigen-decomposition of the Laplacian matrix on each sub-graph provides a rationale to discard eigenmodes with small eigenvalues, which provides a computationally efficient scheme. This also shows that diffusion-based priors are a generalization of conventional Laplacian priors (Penny et al., 2005) and that they are a special case of Gaussian process models (GPMs) that can be inverted using classical covariance component estimation techniques like restricted maximum likelihood (Patterson and Thompson, 1974). This chapter ends with a summary of assumptions used in the current implementation. Chapter 4 is divided into two sections, where we apply the method

to analyze synthetic and real fMRI data. The edge preserving quality of diffusion over a weighted graph is demonstrated first using synthetic data and then applied to real fMRI data. We use three real data sets that include standard ($3mm^3$) and high-resolution ($1mm^3$) fMRI data and report single subject and group analyses. In the discussion we consider general issues, such as the need for explicit spatial models of fMRI given that a well developed framework already exists in SPM, i.e. the mass-univariate approach and RF correction for multiple comparisons, specific assumptions used to implement the method, with emphasis on scalability and future work. Details regarding data sets, standard mathematical results used throughout the thesis and a computationally efficient implementation of the algorithm are given in the appendix.

1 Introduction

Magnetic resonance imaging (MRI) is a non-invasive medical imaging technique used to visualise internal structure and function of the body. Instead of ionizing radiation it uses a powerful magnet to align hydrogen nuclei (protons) in water molecules and radio frequencies to cause them to rotate, which in turn produces a signal that the scanner detects. This signal is manipulated using additional magnetic fields to build an image of structures containing these protons. Functional MRI (fMRI) measures changes in this signal due to neuronal activity, called the Blood Oxygenation Level Dependence (BOLD) response. The exact link between increased activity, evoked by, for example, visual stimulation, and BOLD is still an active area of research (Logothetis and Wandell, 2004; Nair, 2005). Without going into detail, increased oxygen demand leads to a characteristic change in the proportion of oxygenated and de-oxygenated haemoglobin (Hb), which can be measured due to differences in the response of these two forms of Hb. A volume of BOLD responses is acquired at regular intervals, typically every 2-3 seconds, through-out the duration of an experiment, which produces a sequence of scans (volumes) of the brain's response. These data are transformed to a three-dimensional regular grid of voxels (*i.e.* points in the brain) in anatomical space, each containing a univariate observation over time. However, these measurements are noisy and the typical BOLD response is only a small percentage (1-5%) of the global signal, which leads to the following problem statement:

What are the spatial configurations of brain responses to experimental input that best explain a volume of fMRI time-series data? In particular, what are the 'textures' of neuronal responses?

The solution to this requires statistical models, e.g. general linear model (GLM), to explain data and from which inferences can be made as to which regions are active during an experimental condition. One of the most widely used analyses of brain imaging data is Statistical Parametric Mapping (SPM) (Friston et al., 2006), which uses classical statistics in a mass-univariate approach and a random field (RF) correction to deal with multiple comparisons (more on this later). This is a well developed framework, however, the need for models that consider influences among voxels, or multivariate models, stems from the

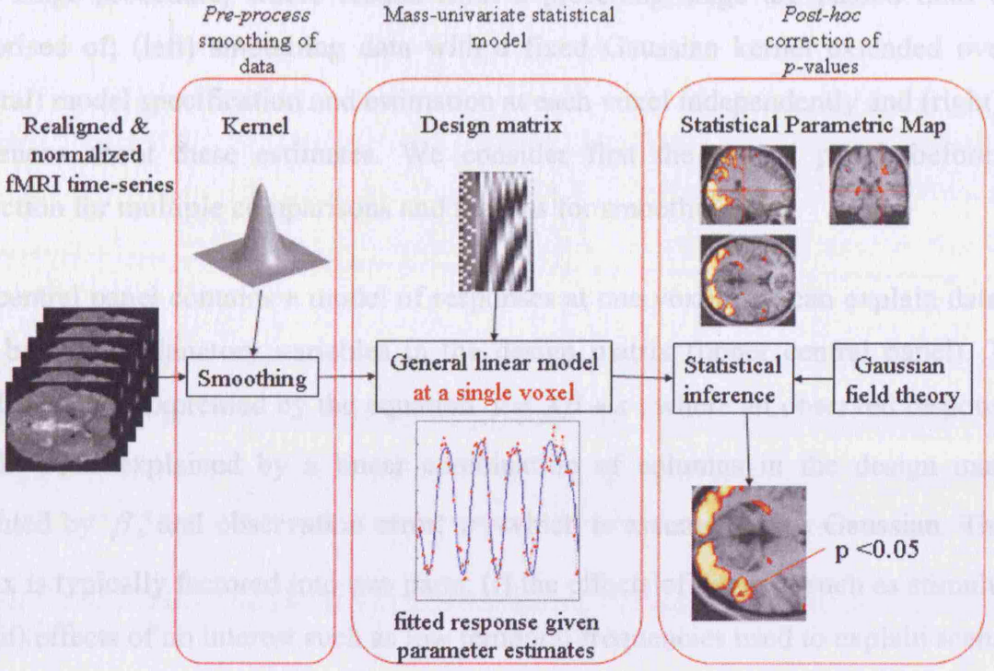


Figure 1-1: Three stage procedure in SPM

The statistical model (central panel) models each voxel separately. Several consequences follow; (i) this statistical model is unable to exploit correlations in measurements over anatomical space and (ii) inferences over many voxels have to deal with spatial dependencies when adjusting for multiple comparisons.

fact that neuroimaging data are generated by spatially extended structures whose spatial scale varies with position in the brain, *e.g.* the organisation of retinotopically mapped responses in visual cortex are segregated into distinct cytoarchitectonic areas with defined boundaries. Despite this, it is currently not possible in SPM to infer whether a model with non-stationary smoothness (*i.e.*, with boundaries) of functionally selective responses is better than a model with stationary smoothness (*i.e.*, without boundaries).

A schematic of the data processing stream in SPM (**http://www.fil.ion.ucl.ac.uk/spm/**) is shown in Figure 1-1 (see (Penny et al., 2001) for an annotated bibliography of methods used in SPM). This is (excluding the pre-process steps of realignment and normalization[1]) a

[1] The main reasons for these pre-processing steps are to (1) remove subject movement artifacts and (2) transform images into a standardized space defined by template images that approximate the atlas of Talairach and Tournoux (Talairach and Tournoux, 1988).

three stage procedure, where results from a preceding stage are passed onto the next, comprised of; (left) smoothing data with a fixed Gaussian kernel extended over space, (central) model specification and estimation at each voxel independently and (right) making inferences about these estimates. We consider first the central panel, before the RF correction for multiple comparisons and reasons for smoothing data.

The central panel contains a model of responses at one voxel that can explain data by, and only by, the explanatory variables in the design matrix (upper central panel). This is a GLM, which is expressed by the equation $y = X\beta + \varepsilon$, where an observed response at one voxel, y, is explained by a linear combination of columns in the design matrix, X, weighted by β, and observation error, ε, which is assumed to be Gaussian. The design matrix is typically factored into two parts; (i) the effects of interest, such as stimulus onsets and (ii) effects of no interest such as low temporal frequencies used to explain scanner drift. The GLM approach is flexible in that it subsumes simpler variants such as the 't-test' for the difference in means to more informed linear convolution models. An introduction to these topics is given in Chapters 2 and 8 on "Statistical Parametric Mapping" and "The General Linear Model" in (Friston et al., 2006).

Linear convolution models (see Chapter 14 on "Convolution Models for fMRI" in (Friston et al., 2006)) are used to represent the BOLD signal, which sums additively [2] to neuronal events at different times. A typical BOLD impulse response, represented using a Haemodynamic Response Function (HRF), characteristically extends over 30 seconds, has a peak response at 6 and undershoots at 16 seconds. Variability in the shape of this response across the brain of an individual and between different people can be accommodated using temporal basis functions. These include the flexible finite impulse response (FIR) and Fourier sets, and an 'informed' basis set (Friston et al., 1998a), based

[2] There is evidence of a nonlinear BOLD response, for example, the saturation effect observed due to short time intervals between stimulus events (known as stimulus onset asynchrony (SOA)) (Friston et al., 1998b; Miezin et al., 2000; Pollmann et al., 1998), where the response to a run of events is less than that predicted by linear summation. This has been found for SOAs below 8 seconds, though for typical SOAs of 2-4 seconds, it is small, i.e. less than 20 per cent (Miezin et al., 2000). These nonlinearities can be modelled using a generalized convolution model, based on the Volterra expansion (Friston et al., 1998b) or the Balloon model (Buxton et al., 1998; Friston et al., 2000) that uses a set of nonlinear ordinary differential equations. However, given the typical SOAs of data used here we assume a linear model.

on a 'canonical HRF', which is characterized by two gamma functions, one for the peak and one for the undershoot, and its partial derivatives [3].

A model of observed responses is specified by convolving the onsets of stimulus events during an experiment with a temporal basis set, e.g. canonical HRF. These form columns in the partition of the design matrix containing explanatory variables of interest. The unknown parameters of the GLM are then estimated. Given P columns in the design matrix, there will be a vector of parameter estimates, $\hat{\beta}$, per voxel of length P. The magnitude of each element of this vector represents the contribution of its associated column in the design matrix to explain the observed response. As this is repeated for all voxels in a brain volume, the result is a volume of estimated parameter vectors, *i.e.* one vector at each voxel, over anatomical space. These can be visualized by taking the same element of this vector from all voxels in a 2D slice. This is a statistical parametric map or SPM that can be overlaid on an anatomical image to show regions of the brain responding to, for example, an experimental stimulus (see top right panel in Figure 1-1). These are often called parameter or 'beta' images as the unknown parameters are typically symbolized by β . This is a static image, as it is comprised of weights of an explanatory variable that extends over time. Combinations of these estimates, called contrasts, can be computed, which are useful for comparing effects of interest, e.g. the difference between responses to famous verses non-famous faces (Henson et al., 2002).

Classical inference proceeds by, for example, comparing the effect size (beta values) with the estimated observation error to form a t-statistic. This can be used to reject the 'null' hypothesis, i.e. that no effect has been measured, if it is above a specified threshold, which is a way to quantitatively protect against false positives. However, because a GLM is applied to each voxel separately and inferences will generally be over many (or a family of) voxels in a volume, there is a multiple comparisons problem. That is, for accurate inference the number of independent observations is required to specify a threshold above which a family-wise 'null' hypothesis can be rejected. A way to address this is to use a Bonferroni correction (see Chapter 17 on "Parametric procedures" in (Friston et al., 2006)), however,

[3] We will use the 'canonical' HRF through-out the thesis, except for the analysis of group data, where we include its first temporal derivative.

this is very conservative and does not consider the spatial nature of data, *i.e.* that an observation at one voxel will typically be correlated with those nearby, which reduces the effective number of independent observations. Spatially correlated random variables are known as random fields (RF), which are used to address this in SPM (Worsley et al., 1996b) (also see Chapter 18 on "Random Field Theory" in (Friston et al., 2006)). Specifically, the smoothness of residuals (i.e. $r = y - X\hat{\beta}$) is estimated (Kiebel et al., 1999) and used to approximate the effective number of independent observations, called RESELS (resolution elements), which are used to select a threshold and correct p-values. This is known as the RF correction.

Intuitively the difference between Bonferroni and the RF correction is that the former controls the expected number of false positives over voxels, while the latter controls the expected number of false positive 'peaks'. If the error is smooth, due to correlation between observations at neighbouring voxels, then there are fewer peaks than there are voxels, which in turn leads to greater sensitivity. As the smoothness of the residuals can be estimated this provides a principled way to chose a threshold and protect against multiple comparisons. Much work has been done on this paradigm, using insights from scale-space theory (Worsley et al., 1996a), generalizing to non-Euclidean spaces (Adler and Taylor, 2007) to detect changes in data projected onto an unfolded, inflated or flattened 2D cortical surface (Taylor and Worsley, 2007; Worsley et al., 1999), thresholding non-stationary SPMs (Taylor et al., 2001) and non-stationary filtering using rotation spaces (Shafie et al., 2003). The result of these efforts is a well developed framework that has been used to detect functional responses in neuroimaging data for well over a decade.

The main reason to smooth data is to improve the signal-to-noise ratio. In particular, it can be motivated by the following four points (see Chapter 2 on "Statistical parametric mapping" in (Friston et al., 2006)). The matched filter theorem states that a signal, e.g. activated brain region, can best be recovered from noisy data by smoothing it with a kernel of the same size. The spatial scale of the HRF is ~3-5mm (Nair, 2005). Smoothing data makes the distribution of errors more normal, which improves the validity of inferences based on parametric tests. In particular, when making inferences over a volume, the RF correction assumes the estimated observation error at discrete points in the brain, *i.e.*

voxels, approximates an underlying continuous RF, which is improved if its smoothness is greater than voxel size, e.g. FWHM greater than ~3 voxels.

The mass-univariate (with RF correction) approach is a principled framework for model specification, estimation and inference given spatio-temporal data. However, spatial properties (*i.e.* that necessarily include more than one voxel) of neuronal responses are not included in the model (central panel); they are considered before and after modelling *per se*. Given the importance of spatial correlations in this framework, it seems reasonable to include them in the modelled response at a single voxel (central panel), that is; data at one voxel is no longer only explained by explanatory variables in the design matrix, but also by responses of its neighbours, where the size and shape of this neighbourhood characterises spatial correlations in the data.

Another perspective on the three stage procedure shown in Figure 1-1 is that spatially correlated fMRI data cannot be generated from this model, as there are no spatial parameters. As such it is not a generative model (Bishop, 2006) of spatially distributed changes in BOLD. This may seem trivial; however, it reflects a deep issue: in order to test a hypothesis, a data model has to be formulated, which can generate features that are salient to that hypothesis (*e.g.*, temporally structured activity in spatially segregated and functionally selective brain regions). Given this, a prior over GLM parameters (and observation error) can be specified that encodes spatial dependence. The benefit of having an explicit spatial model of GLM parameters is that the three stage procedure can be subsumed into one generative model. This allows comparison of different generative models in order to quantify which of these has an optimal balance between accuracy and complexity (details on the Bayesian approach to data analysis will be given in the following section). The challenge for requisite *multivariate* models is to embody the general organizational principles of functional segregation and integration (Friston, 2002) into *spatial* models of how data are generated.

This has led to the development of several Bayesian approaches to spatial models of fMRI data, which include stationary Gaussian Markov Random Field (GMRF) priors (Descombes et al., 1998; Gossl et al., 2001; Penny et al., 2005; Woolrich et al., 2004), multi-resolution

MRF priors based on wavelets (Flandin and Penny, 2007) and kernel-based methods (Harrison et al., 2008). The benefit of a Bayesian framework is that the evidence for different spatial priors can be compared (MacKay, 2003) and as priors encode hypotheses about how we think data are generated, competing hypotheses can be compared quantitatively (Penny et al., 2007). A typical GMRF prior encodes local dependence between voxels using a Laplacian or bi-Laplacian precision matrix (these terms will be explained in more detail later). A consequence of using a MRF prior is to incorporate local averaging (*i.e.* smoothing) into a generative model of the data, which can be estimated for each parameter image according to a measure of uncertainty in that parameter. This means that a parameter estimate, at a particular voxel, will be more similar to its nearest neighbours if there is evidence for a smooth response to its associated explanatory variable. Note the important difference here compared to smoothing data, which effectively smoothes all parameter images to the same degree by an amount that is chosen by the user and not estimated from data. An issue with using a stationary GMRF prior is that given the convoluted nature of grey matter and patchy functional segregation, a non-stationary spatial model, where the degree of smoothness can depend on spatial location, may be required to model spatial features optimally. A step in this direction has been the use of the multiscale properties of wavelets (Flandin and Penny, 2007); however, basis functions that adapt, given local geometric information may provide a more general framework. This lead to a recent proposal based on diffusion kernels on arbitrary graphs (Harrison et al., 2007a; Harrison et al., 2008), which is the topic of this thesis. Although we consider only the simplest noise model in this thesis, more realistic models in the literature include stationary spatiotemporal autoregressive models (Penny et al., 2007; Woolrich et al., 2004).

Non-Bayesian approaches include a number of proposals, applied to anatomical and functional MRI data, that use techniques from image processing (Aubert and Kornprobst, 2002; Romeny, 1994). Approaches based on nonlinear diffusion have been applied to MRI data (Gerig et al., 1992) and the Laplace-Beltrami operator (a generalization of the Laplace operator to a Riemannian manifold) used in a statistical approach to deformation based morphometry (Chung et al., 2003). Related work using the eigensystem of a finite element approximation to the Laplace-Beltrami Operator has been used to smooth structural and

fMRI data (Qiu et al., 2006) and its diffusion kernel to model cortical thickness and density (Chung et al., 2007). Similarly, nonlinear diffusion (Hollander and Bajla, 1998; Kim and Cho, 2002; Kim et al., 2005; Sole et al., 2001) and the bilateral filter (Polzehl and Spokoiny, 2001; Smith and Brady, 1997; Tabelow et al., 2006; Walker et al., 2006) have been used to adaptively smooth functional images and a general framework proposed for both anatomical and functional images (Faugeras et al., 2004). Graph-based diffusion has been used to regularize diffusion tensor images (DTI) (Zhang and Hancock, 2006), where a weighted graph Laplacian (the discrete analogue of the Laplace-Beltrami operator) was used to adaptively smooth a field of diffusion tensors, thereby preserving boundaries between regions, *i.e.* white matter tracts. An alternative to 3D spatial models is a 2D surface based approach, where fMRI data are projected onto a cortical mesh constructed from a structural MRI for visualization (Memoli et al., 2004; Teo et al., 1997) or performing the statistical analysis (Andrade et al., 2001; Kiebel et al., 2000). Other approaches to adaptive analysis of fMRI include Canonical Correlation Analysis (Friman et al., 2003) and using spectral clustering to divide a volume of fMRI data into homogeneous patches or parcels (Flandin et al., 2002; Thirion et al., 2006).

Despite this, the mass-univariate approach remains the conventional practice. However, with increased use of high-resolution fMRI (hr-fMRI), where identifying the 'texture' of neuronal responses is important, the demand for explicit spatial models of fMRI data is likely to grow. It is one of the aims of this thesis to set out a framework that combines many core ideas currently implemented in SPM with techniques used in image processing and Bayesian models of spatial data, which can used to represent the texture of neuronal response. In the next subsection we will describe the relevant background on the fundaments of our proposed solution.

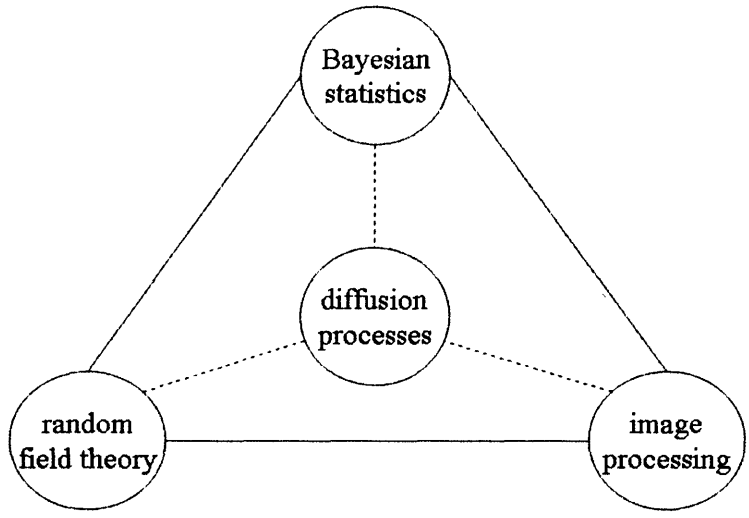


Figure 1-2: Fundaments of our proposed solution

The solution proposed in this thesis relies on links between ideas in Bayesian statistics, image processing and random field theory. In particular, how they relate to diffusion processes.

1.1 Fundaments of a solution

The current work draws on three main sources in the literature; Bayesian statistics (Bishop, 2006; MacKay, 2003), in particular, Gaussian process models (GPM) (MacKay, 1998; Rasmussen and Williams, 2006) used in machine learning, diffusion-based methods in image processing (Aubert and Kornprobst, 2002; Chan and Shen, 2005; Scherzer et al., 2008), specifically using a graph-based formulation (Zhang and Hancock, 2005; Zhang and Hancock, 2007) and random fields (Adler, 1981; Bishop, 2006; Geman and Geman, 1984), examples of which are GMRF and Gaussian process priors (GPP). The literature on each topic is huge; however, a common theme, relevant to this thesis, is the process of diffusion. This can be used to represent *parameter values* of a GLM, as a general, multi-dimensional random field over anatomical space, where the process of diffusion represents spatial dependence between voxels in a hierarchical model. The purpose of this subsection is to provide some background on each of the topics above, with emphasis on intuition and links between them and to diffusion processes (see Figure 1-2).

1.1.1 Bayesian statistics

The aims of this subsection are to motivate a Bayesian approach to fMRI data and distinguish it from a closely related approach based on regularization. In particular, we want to distinguish the maximum a posteriori (MAP) estimate from an estimation scheme based on the model evidence. This reveals the benefit of integrating out uncertainty in a model and how it leads to a principled way to approximate hyper-parameters of a model and select among models based on their accuracy and complexity. It also reveals similarities between spatial regularizers employed by, for example, the Energy Method (Aubert and Kornprobst, 2002) used in image processing and GMRF and GP priors used in Bayesian formulations of spatial statistics.

The Bayesian paradigm, in particular the use of hierarchical models, is at the heart of empirical Bayesian methods used in the analysis of neuroimaging data (Friston et al., 2002a; Friston et al., 2002b). Their appeal is that they provide an intuitive and easily implemented scheme to learn (empirical) priors, given data. The central idea is that a prior over model parameters can be optimized (or learnt) through further constraints at a higher level. This leads to an observation model comprised of levels, or a hierarchy, where each level provides constraints for the one below. Upward and downward passes of sufficient statistics enables learning of priors, given data and as such are called *empirical* priors.

A simulated volume of brain data is obtained by sampling from the probability density induced by a hierarchical model. A graphical representation of the generative and implicit recognition models used in this thesis are shown in Figure 1-3. Considering the generative model (left) first, nodes and arrows represent random variables and conditional dependence respectively. The model, m, represents the structure and form of the probability densities of the graph, which is a hypothesis of how data are generated. Parameters of a model, β, weight *temporal* explanatory variables contained in a design matrix, as described earlier. These encode experimental conditions such as stimulus onsets. The GLM at each voxel contains a vector of parameters resulting in a field of vectors over anatomical space. A crucial difference compared to the mass-univariate approach is that hyper-parameters, α, control the density over these parameters *e.g.* its spatial smoothness. These models can

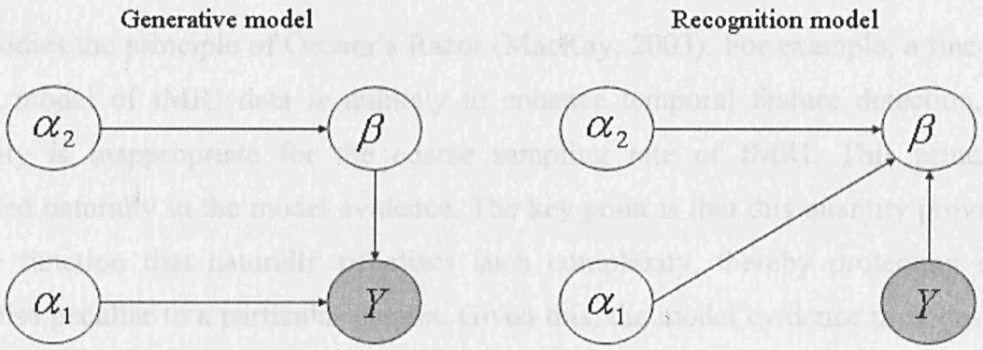


Figure 1-3: Generative and recognition models

Graphical representation of a generative (left) and recognition model. Each node represents a random variable, where the observed variable, *i.e.* data, is shaded and arrows indicate conditional dependence.

generate synthetic data that contain features similar to those observed in real data, e.g. spatially correlated observations over time. By 'reversing' the arrows we can use this representation of a generative process to *recognize* the model in data. An example of this is shown on the right of Figure 1-3.

The aim of recognition is to compute a hierarchy of posterior densities over parameters, hyper-parameters and the model itself. The posterior over parameters encodes not only the most likely response, over anatomical space, but also a *measure of uncertainty* about the parameters, given data. This means that an ensemble of parameter estimates is represented by the posterior density instead of just one solution. This is important because the true parameter values are not observed and therefore we can, at best, only have a degree of belief as to what they truly are given data. This probability density can be used to identify patterns of response using posterior probability maps (PPMs) (Friston and Penny, 2003), which are used to visualize structure-function relationships that include a measure of uncertainty after the model has been optimized [4]. Importantly, a density over hyper-parameters, which controls the degree of uncertainty in parameters, is also estimated. This leads to two quantities (considered in more detail shortly) referred to as the evidence of the hyper-parameters and the model evidence. Within a Bayesian paradigm, the intuition is that data are best explained using an optimal balance between model accuracy and complexity

[4] These involve thresholding the posterior density to produce a map that represents regions of anatomical space where the probability of parameter values above a threshold have a specified degree of certainty, *e.g.* regions that have parameter values above zero with probability greater than 0.95. PPMs are important, as they are the basis for inference and hypothesis testing.

that embodies the principle of Occam's Razor (MacKay, 2003). For example, a fine-scaled temporal model of fMRI data is unlikely to enhance temporal feature detection, as its complexity is inappropriate for the coarse sampling rate of fMRI. This principle is represented naturally in the model evidence. The key point is that this quantity provides an objective function that naturally penalizes such complexity, thereby protecting against fitting noise peculiar to a particular dataset. Given this, the model evidence provides a way to compare different models, with varying degrees of complexity and enables us to formalise the question, "which model do our data support?" using *Bayesian model comparison*. Bayesian spatiotemporal models therefore allow us to compare models with and without spatially coherent responses and ask whether this coherence is stationary (*i.e.*, the same over space) or not. This sort of inference is central to asking questions about the nature of functional segregation in the cortex, or indeed sub-cortical structures.

We consider these ideas in more detail next, with emphasis on differences between regularization, the MAP estimate and a model evidence based framework.

1.1.1.1 Relation to regularized solutions

The links between classical approaches, such as ordinary least squares (OLS) and penalized least squares (PLS) estimates and formulations based explicitly on probabilities, such as maximum likelihood (ML) and maximum a posterriori (MAP) estimates are well known (Bishop, 1995) (also see (Tipping, 2004) for an introduction), however, it is instructive to look at two simple examples that we will refer to later. The examples are simple in that the first considers temporal data only, e.g. from one voxel, while the second involves data from a 2D scalar image, i.e. spatial data only, in which case the design matrix is a scalar equal to one.

Given a univariate time series of data comprised of T observations, y, and a model, $y = X\beta$, where X is the design matrix containing P columns of explanatory variables, e.g. basis functions, the OLS estimate of the unknown parameters, $\hat{\beta} = (X^T X)^{-1} X^T y$, is obtained by minimizing the sum-of-squares error (SSE) function, $E_D = \frac{1}{2}(y - X\beta)^T (y - X\beta)$. A well known problem with this is that it can lead to over-

fitting of the data, that is; fitting noise as well as the underlying signal, which leads to poor generalization, i.e. using $\hat{\beta}$ to fit independent data (that was not used to estimate β) results in a poor fit. A way to address this is to add an additional term, which is typically taken as the squared sum of the weights, i.e. $E_W = \frac{1}{2}\beta^T\beta$, and scaling it by a regularization constant λ. This leads to the estimate, $\hat{\beta} = (X^T X + \lambda I_P)^{-1} X^T y$, where I_P is the identity matrix, which minimizes the objective function,

$$E(\beta) = E_D(\beta) + \lambda E_W(\beta) \qquad 1.1$$

where the regularization constant balances the trade-off between how well the estimated response fits the data and how smooth it is.

Both these estimates can be reformulated in terms of probabilities, where each data point is thought of as sampled from a Gaussian density (see Appendix II D). Given the GLM, $y = X\beta + \varepsilon$, this is equivalent to assuming the error at each observation, y_i, is a sample from the Gaussian density $p(y_i \mid X_i, \beta, \alpha_1)$ which has precision (inverse variance) α_1. The probability of observing the vector y is then given by the product $p(y \mid X, \beta, \alpha_1) = \prod_{i=1}^{T} p(y_i \mid X_i, \beta, \alpha_1)$ and the MLE is the value of β that maximizes this probability. Taking the negative logarithm of this expression reveals a term that is proportional to the SSE function above. Prior knowledge as to the value of β can be introduced by including a prior density, $p(\beta \mid \alpha_2)$, which is typically chosen as Gaussian. Bayes rule is then used to compute the posterior density over parameters, after observing the data, which is

$$p(\beta \mid y, \alpha_1, \alpha_2) \propto p(y \mid \beta, \alpha_1) p(\beta \mid \alpha_2) \qquad 1.2$$

We will consider the normalization constant of this density later, as the most probable value of β, i.e. that which maximizes this expression, does not depend on it. Typically the logarithm of this expression is used for optimization, which leads to the objective function

$$E_{MAP}(\beta) = -\log p(\beta \mid y, \alpha_1, \alpha_2)$$
$$= \tfrac{1}{2}\left(\alpha_1 (y - X\beta)^T (y - X\beta) + \alpha_2 \beta^T \beta\right) + const \qquad 1.3$$

Which has the same form as for the objective function that lead to the PLS estimate, with $\lambda = \alpha_2 / \alpha_1$.

A common approach to choosing a value for λ is cross-validation, where errors calculated using independent data are used to assess a good value, however, this involves dividing the data into training and test sets. The main point here is that the MAP estimate is often described as a Bayesian approach in that it is formulated in terms of probabilities; however, it overlooks a distinguishing feature of Bayesian methods, which is to integrate out uncertainty in the model parameters. A consequence of this is an objective function that automatically penalizes overly simple and complex models, which can be used to estimate hyper-parameters, α, and approximate the probability of the model itself, as we outline next.

Integration over parameters, β, leads to the *marginal likelihood*

$$p(y \mid \alpha) = \int_b p(y \mid \beta, \alpha_1) p(\beta \mid \alpha_2) \qquad 1.4$$

which is the normalization constant referred to above [5]. In the Gaussian case this integral has a known form, which can be used to optimize the hyper-parameters. This is typically called the type-II maximum likelihood estimate (ML-II). An advantage of this approach over cross-validation is that all data are used to estimate hyper-parameters. An additional benefit of computing the quantity in Eqn 1.4 is that it can be used to compare models.

To see this, we consider the posterior density in Eqn 1.2 as the first level in a hierarchy of posterior probabilities, where we include another variable, m, to represent a specific model. The posterior of the hyper-parameters can be written using a prior over them,

[5] also known as the evidence of the hyper-parameters

$p(\alpha \mid m)$, times the marginal-likelihood in the numerator (i.e. it plays the same role as the likelihood in the posterior over parameters),

$$p(\alpha|y,m) = \frac{p(y \mid \alpha,m)p(\alpha \mid m)}{p(y \mid m)} \qquad 1.5$$

The denominator of this expression is known as the model evidence. The integral required to compute this is typically intractable and requires an approximation, e.g. Laplace's method about the most probable values of the hyper-parameters (MacKay, 2003). For details on computing the approximate log model evidence see Appendix II E.

Similarly, the model evidence can be used to formulate the posterior probability of the model

$$p(m|y) = \frac{p(y \mid m)p(m)}{p(y)} \qquad 1.6$$

Typically the prior over different models is taken to be uniform, which means that the posterior of the model is proportional to the model evidence, i.e. $p(m|y) \propto p(y \mid m)$.

The Bayesian framework not only offers the most probable estimate of the parameters, but a hierarchy of posterior densities that includes the hyper-parameters and the model itself, which is not possible using cross-validation. The consequence is a framework that can be used to select between different explanations, i.e. hypotheses, as to how the data were generated. This can be achieved by computing the ratio of two models, $p(m_1 \mid y)/p(m_2 \mid y)$, which is known as the Bayes factor, where a ratio greater than 100 is considered decisive evidence in favour of m_1 (Kass and Raftery, 1995). We will use this in Chapter 4 to compare different spatial models of fMRI data.

Our second example considers data, Y, i.e. a column vector comprised of N observations, e.g. containing the grey-scale values of pixels in an image, expressed by the equation $Y = \beta + \varepsilon$. Following the same reasoning as above, a regularized solution can be found by

considering an additional term, e.g. $E_W = \frac{1}{2}\int_\beta |\nabla\beta|^2$, where $\nabla\beta$ is the spatial gradient of the continuous function β. This is known as Tikohonov regularization (Aubert and Kornprobst, 2002) and leads to a solution by minimizing an objective function analogous to Eqn 1.1. It is instructive to consider the discrete analogue of this additional term, using the matrix A to denote the approximate gradient operator, which leads to $E_W = \frac{1}{2}(A\beta)^T(A\beta) = \frac{1}{2}\beta^T Q\beta$, where $Q = A^T A$. This matrix can take many forms as long as it is positive semi-definite.

Reformulation in terms of probabilities leads to the MAP estimate by maximizing $\hat{\beta} = \arg\max_\beta p(Y \mid \beta, \alpha_1) p(\beta \mid \alpha_2)$, where now $p(\beta \mid \alpha_2)$ is a spatial prior. A typical example of the joint distribution of a GMRF prior is

$$p(\beta \mid \alpha_2) = (2\pi)^{-N/2} \det(Q)^{1/2} \exp(-\alpha_2 E_W(\beta)) \qquad 1.7$$

where Q is known as a precision matrix. Typical examples include the Laplacian, L, which we will review in detail in Chapter 2, and the bi-Laplacian matrix, i.e. $Q = L$ and $Q = L^2$ respectively.

Gaussian process priors provide another example, where the prior is now of a continuous function over space, whose density is prescribed by a mean and covariance *function*, e.g. the squared exponential function (described later). A sample from a finite set of points can be computed using the covariance function to produce a covariance matrix, K, in which case the distribution of this set of random variables will be Gaussian, represented by Eqn 1.7, but with $Q = K^{-1}$.

GMRF and GP priors play the same role as the regularizer, in that non-zero off-diagonal terms in Q couple random variables over space, which in turn induce smoothness of samples generated from them. However, the crucial difference with regularization is that because the model is framed in terms of probabilities, the marginal likelihood and model evidence can be used to go beyond computing only the MAP estimate to one that can

quantitatively evaluates the evidence for hyper-parameters and compare different models. We will return to GMRF and GP priors in the subsection on "Random fields".

As Bayesian models are based on probabilities they are intimately related to diffusion. The simplest example of this is the Gaussian density, which is the solution of the heat equation on the real line. This idea extends to other domains, such as the circle and in general Lie groups (Brockett, 1997). Next we consider the role of diffusion in image processing.

1.1.2 Image processing

The aim of this subsection is to provide a brief overview of some of the techniques used in image restoration[6] and segmentation, which can be used as a resource of spatial models for neuroimaging data. We will consider approaches based on (1) minimizing an energy function (or functional in the continuous case) and partial differential equation (PDEs) and (2) continuous and discrete (i.e. graph-based) formulations. We have chosen to use a graph-based approach in this thesis, because it provides a general way to generate kernels over 3D or 2D embedded spaces, e.g. cortical mesh, from sparse matrices. Graphical models in machine learning (Jordan, 1999) also provide a general formulation of statistical models. The similar benefits of graph-based diffusion methods in image processing further motivates the use of graph-theoretic approaches to represent and estimate statistical images of neuroimaging data. Image segmentation, in particular graph-partitioning algorithms (Grady and Schwartz, 2006; Qui and Hancock, 2007; Shi and Malik, 2000), is included as it can be used to reduce the graph-Laplacian defined over the whole brain volume to a block diagonal form. The primary motive for this is to reduce computational burden.

Early work in image processing can be broadly divided into methods that minimize an energy functional and partial differential equations (PDEs). The Energy Method (Aubert and Kornprobst, 2002) conveniently links to regularization theory and MAP estimates,

[6] The process of restoration is to "...remove or diminish the effects of [image] degradation." (Aubert and Kornprobst, 2002), where an image can be degraded through either a deterministic or stochastic process, for example due to the mode of image acquisition or noise in the measurement device. In contrast, an example of image enhancement is to deblur an image. The distinction here is between algorithms that smooth an image using forward diffusion, which does not create any new edges and those that enhance edges, which can be achieved using backward diffusion, shock filters or PDEs with reaction terms (Zhu and Mumford, 1997).

described in the previous subsection. It can be considered as a static problem, where we are given a source term and the objective is to find the image that minimizes an energy functional, i.e. the restored image. The solution is regularized using an additional term, where a typical example is Tikhonov regularization (briefly described earlier), which is based on the squared norm of the gradient. This can be generalized to a *function* of this quantity for anisotropic smoothing. A classic example of which is the Total Variation (Rudin et al., 1992), which uses the absolute value of the gradient instead. An alternative approach is to formulate the problem in terms of a system of PDEs, which we consider next.

A main difference with the PDE-based approach is that the problem is now considered as an initial value problem, where the image is the initial condition and a PDE is used to propagate a solution to another point in time. A sequence of images is produced through this process, where as time increases the image is simplified, or smoothed, and is the reason why time is referred to as a scale variable. In other words a smoothing PDE can be thought of as a low-pass filter. This is now a dynamic problem in contrast to the Energy Method. The type of PDE used depends on the task, for example image restoration, which we consider next.

The classic example of a restorative PDE is the heat (diffusion) equation, which the image processing community have used for many years (Knutsson et al., 1983; Koenderink, 1984; Witkin and Witkin, 1984). For overviews, from the perspective of scale-space theories see (Lindeberg, 1994; Romeny, 1994; Romeny, 2003) where an image is represented as a one-parameter family of smoothed images, which is parametrized by the size of the smoothing kernel used to suppress fine-scale structures. This scale space representation can be generated by the diffusion equation, which describes the density fluctuations in an ensemble undergoing diffusion, i.e. the 'classical heat equation'; $\dot{f} = c\Delta f$, where the Laplace operator (second-order spatial derivative), Δ, is the composition of the gradient and divergence operators, i.e. $\Delta = div(grad)$, f is regarded as the density of the ensemble (*e.g.*, image intensity) and c is a constant diffusion coefficient.

A typical use in image processing is to de-noise an image, where the noisy image is the initial condition, $f(t=0)$ and a smoothed, de-noised, image is the result of integrating the heat equation to evaluate the diffused image at some time later; $f(t)$. An important property of solutions to the heat equation is that no spurious details are generated through the fine to coarse representation (Koenderink, 1984; Witkin and Witkin, 1984). Perona and Malik (Perona and Malik, 1990) contributed by using nonlinear diffusion models to preserve the edges of images using an image dependent diffusion term, $c = c(\nabla f)$, where now the diffusion equation takes the form, $\dot{f} = \nabla \cdot c(\nabla f)\nabla f$. The dependence of the diffusion coefficient on the spatial gradient of the image has the effect of reduced diffusion over regions with high gradient; *i.e.* edges of the image. Under certain conditions this produces forward diffusion that smoothes homogeneous regions while preserving edges [7].

Later formulations of nonlinear diffusion methods include using directional information (Weickert, 1998), where the function $c(\nabla f)$ is replaced by the tensor $D(\nabla f)$, i.e. $\dot{f} = \nabla \cdot D(\nabla f)\nabla f$ and the general framework of (Alvarez et al., 1992), which leads to a natural generalization of linear scale-scale. Of relevance to the approach adopted in this thesis are those that consider an image as a surface embedded in a high dimensional space (Kimmel, 2003; Sochen et al., 1997). This lead to a general framework based on the Laplace-Beltrami operator (LBO), which is the generalization of the Laplace operator to a non-Euclidean, i.e. curved, space (Rosenberg, 1997). The convenience is that their scheme can easily be extended to vector-valued images on arbitrary surfaces, for example a colour image on a sphere. The discrete analogue of the LBO is the weighted graph-Lapalcian (Chung, 1997). This has been used to generate a graph scale-space representation of an image, which has been employed to adaptively smooth scalar, vector and matrix-valued images (Zhang and Hancock, 2005).

Turning now to image segmentation, examples using a continuous representation include early work identifying edges using sharp variations in image intensity (Canny, 1983;

[7] However, if these are not met it yields backward diffusion. This can be used to enhance edges, e.g. deblur an image, however, we do not use this class of PDEs in this thesis. Note that the nonlinear diffusion equation can be regularized by smoothing the gradient of the image (Catte et al., 1992).

Prewitt, 1970; Roberts, 1965; Sobel and Feldman, 1973). Further developments included minimizing an energy functional to find a set of nearly piecewise constant approximations to an image and its edges (Mumford and Shah, 1989) and "active contours" (Osher and Paragios, 2003), where a closed curve around an object is shrunk until it reaches the object boundary in the image. Discrete formulations include graph-partitioning algorithms, e.g. Chaco (Hendrickson and Leland, 1994), Metis (Karypis and Kumar, 1998), Meshpart (Gilbert et al., 1998) and Graph Analysis Toolbox (Grady and Schwartz, 2003). Many of these methods use the graph-Laplacian (Chung, 1997). In particular, spectral graph partitioning (Shi and Malik, 2000) uses the second eigenvector (Fiedler vector) of the graph-Laplacian, which can be repeated for each segment, in a recursive scheme. This requires computing eigenvectors, which for large graphs; e.g., with greater than $10^{5\text{-}6}$ vertices, is a computational challenge. Alternatives that also use the graph-Laplacian are based on commute times (Qui and Hancock, 2007) or the isoperimetric approach (Grady and Schwartz, 2006), which uses the solution of a linear system of equations instead of the Fielder vector to partition a graph. Given the large number of voxels in a brain volume we chose this latter approach over spectral graph partitioning.

1.1.3 Random fields

The purpose of this subsection is to provide some intuition about RFs, their relation to GPPs, a diffusion process and GMRF priors. A 'random field' refers to a collection of random variables, typically, over more than one dimension. They can be continuous, i.e. an infinite set where any two points can be infinitesimally close or a finite (discrete) set. The notion of a scalar random field can be extended to multiple dimensions, where one or more numbers describe the field at each point in space, *e.g.* flow. Generalizing further, the field can be on a curved surface, *e.g.* temperature fluctuations on the two-dimensional surface of a curved object. This is an example of a continuous random field on a non-flat, i.e. non-Euclidean, space, which in general is called a manifold. We will first consider continuous random fields before their discrete analogues.

A Gaussian process prior (GPP) is a continuous random field that is used within a Bayesian framework to constrain the estimation of parameters in an observation model *e.g.*

autocorrelation functions over time or GLM parameters over space in a brain volume. A GPP is an infinite collection of random variables, any finite number of which have a joint Gaussian distribution (MacKay, 2003; Rasmussen and Williams, 2006). As such it is completely specified by a mean and covariance function, which can take many forms, as long as it is positive semi-definite. This is a very flexible prior as it is a prior over a function, which can be used to model general data, not just images and provides (exact) analytic solutions. They are easily generalized to model non-Gaussian processes through specifying a transformation, *e.g.* log-transform to model a random field of strictly positive numbers, which have been referred to as a 'warped' GPP in the machine learning literature (Snelson and Ghahramani, 2007). Generalizing this notion further, a GPP can be defined on any arbitrary surface (sub-manifold), *e.g.* a cortical surface.

Diffusion occurs due to the random motion of 'particles' in a field, *e.g.* molecules in air, and is an example of a local Gaussian process. This process is described by the heat or diffusion equation, whose solution is given by the diffusion kernel, which propagates a function over space from one moment to the next. In other words, the diffusion kernel defines what the function will be at a later time. If the diffusion process is over physical space it contains spatial information and can be used as the spatial covariance of a probability density, thereby providing a representation of random field.

Diffusion methods in image processing and covariance functions in GPMs provide the means to represent a function over space; however, the emphasis of each approach is different. One main difference is that a GPM is a statistical model from which inferences and predictions can be made (MacKay, 1998). The objective is not solely to smooth data, but to estimate an optimal smoothing operator, which is used to represent the spatial process from which data are generated. The relation between models of diffusion and GPPs is seen when we compare the diffusion kernel of the classical heat equation and the squared exponential (SE) covariance function typically used in GPMs (Rasmussen and Williams, 2006). Note that there are many other covariance functions that can be used (Abrahamsen, 1997). In two dimensions, (x_k, x_l), where subscripts indicate location in the domain and c is a scalar. The heat equation, its solution and diffusion kernel are

$$\begin{aligned} \dot{f} &= c\Delta f \\ f(x,t+\tau) &= \int_{x'} K(x,x';\tau) f(x';t) \\ K(x_k,x_l;\tau) &= \frac{1}{4\pi c\tau}\exp\left(-\frac{(x_k-x_l)^T(x_k-x_l)}{4c\tau}\right) \end{aligned} \qquad 1.8$$

where $K(x,x';\tau)$ is the diffusion kernel that represents the solution of the heat equation. The first line is the special case of constant diffusion coefficient. The solution of this equation is given in the second line, where the image at time t, $f(t)$, is propagated to $t+\tau$ by convolution with the diffusion kernel, shown in the last line[8]. This is Gaussian with variance $2c\tau$, meaning that the image at $t+\tau$ is a smoothed version of $f(t)$. Typically, a GPP has an additional scale hyper-parameter to give

$$K(x_k,x_l;\alpha) = \upsilon\exp\left(-\frac{(x_k-x_l)^T(x_k-x_l)}{2\sigma^2}\right) \qquad 1.9$$

where $\alpha = (\upsilon,\sigma)$. This has the same form as the diffusion kernel above where the squared characteristic length-scale is $\sigma^2 = 2c\tau$, i.e. σ^2 increases with time τ . A zero mean GPP is then specified, at a set of locations, by the multivariate distribution $p(f) \sim N(0,K)$ (Rasmussen and Williams, 2006).

Discrete random fields, e.g. GMRF have been used in image processing (Geman and Geman, 1984; Li, 2001) and neuroimaging, e.g. MEG/EEG and fMRI (Descombes et al., 1998; Gossl et al., 2001; Penny et al., 2005), as pointed out earlier. Diffusion in a continuous media has a discrete analogue on a graph (Chung, 1997) that is comprised of a set of nodes and weighted edges. The Laplacian of a graph is computed using the edge weights between nodes and the diffusion kernel is obtained from the matrix exponential of the Laplacian[9] (see Eqn 2.33). A typical example of a GMRF prior is where the precision matrix is the Laplacian matrix mentioned earlier. However, given the connections between diffusion kernels and GPPs above, the diffusion kernel on a graph can also be used as the

[8] Element-wise exponential is used in this line as opposed to matrix exponential used later for the diffusion kernel on a graph

[9] The convolution in Eqn 1.8 is then the matrix-vector product

covariance matrix of a spatial prior, which is an example of a diffusion-based spatial prior. The benefit over a GMRF with precision matrix given by the Laplacian matrix is that the size (and shape) of a voxel neighbourhood can also be optimized. In this thesis, we use this form of spatial prior over GLM parameter images.

1.2 Final remarks

The contribution of this work is to combine methods from image processing and Gaussian process models from machine learning to provide an explicit spatial model of fMRI data based on random fields. The motivation is to incorporate smoothness into the statistical model by making it a hyper-parameter of the model and estimating it using empirical Bayes. This uses random fields, similar to SPM, but in a different way, i.e. to represent spatial dependencies between parameter vectors at different voxels (and measurement error), instead of correcting mass-univariate statistics for multiple comparisons. A benefit of doing this is that parameters and hyper-parameters used in the three-stage procedure in SPM are a function of just one objective function. This can be used to optimise spatial dependencies between parameter estimates, allows one to infer the presence of spatially organised responses and has the potential to greatly improve spatial feature detection. Critically, this work provides a hypothesis-driven framework; in that a formal model embodies a hypothesis about how we think data are caused. This is important as we develop models that explicitly include spatiotemporal aspects of functional and anatomical principles.

The potential benefits of this approach are far reaching in that it promises to answer questions, with a measured degree of certainty, about the 'texture' and 'shape' of functional responses. These questions are becoming increasingly important in imaging neuroscience, for example, investigating midbrain structures such as the periaqueductal gray (Mobbs et al., 2007) in anxiety-related disorders, superior colliculus (Schneider and Kastner, 2005; Sylvester et al., 2007), retinotopic maps of the visual cortex (DeYoe et al., 1994; Engel et al., 1994; Sereno et al., 1994; Warnking et al., 2002) and lateral geniculate nucleus (Haynes et al., 2005), and the fine functional structure within fusiform face area (Grill-Spector et al., 2006). This last example is important as the correspondence that followed this paper

indicated that the simple rules used to evaluate the 'texture' of response were not correctly formulated, leading to serious criticism of some of their results (Baker et al., 2007; Simmons et al., 2007). A more suitable analysis would be one that models explicitly the spatial features, or geometries, of neuronal responses we want to make inference about.

In the next chapter we review topics in graph theory required to formulate diffusion kernels on graphs. These will be used in later chapters as the spatial covariance matrix of a prior over parameters, i.e. beta images, of a GLM of fMRI data.

2 Theoretical background

Much of the current work relies on several results from graph theory, in particular (1) using the Laplacian matrix or graph Laplacian (GL) to represent a graph, (2) the edge weights of a graph, which encodes its topology, *i.e.* connectivity, (3) its eigensystem, (4) the diffusion (or heat) kernel of a GL, and (5) using the GL to partition a graph, *e.g.* brain volume, into computationally manageable segments. The purpose of this chapter is to review this material for graphs in 2D, which we extend to 3D in Chapter 4. Even though the algorithm developed in this thesis uses a Laplacian defined on a finite graph, useful insights can be gained by considering continuous formulations. This leads to thinking of images as *surfaces* embedded in a higher dimensional space and the Laplace-Beltrami operator (LBO) used to represent diffusion over them (Aubert and Kornprobst, 2002; Kimmel, 2003; Sochen et al., 1997). We will review this material in subsection 2.2.

2.1 The Graph Laplacian

We consider an undirected graph with vertices (nodes) and edges, which we denote by $\Gamma = (V, E)$. The vertex and edge sets are V and $E \subseteq V \times V$, respectively. An element of each is $v_i \in V$ and $e_{ij} \in E$, where an edge connects two neighbouring vertices v_i and v_j, denoted by $i \sim j$. In general there is a weight associated with each edge, w_{ij}, which are symmetric as the graph is undirected, i.e. $w_{ij} = w_{ji}$. The total number of nodes and edges are $N_V = |V|$ and $N_E = |E|$, where the vertical bars indicate cardinality, *i.e.* number of elements in a set. Here we consider graphs with equally spaced nodes with nearest neighbour topology, i.e each node is coupled to its four closest neighbours, except at boundaries. Examples of other network topologies used in image processing are unequally spaced nodes (Grady and Schwartz, 2003) and small world networks (Grady and Schwartz, 2004).

The graph Laplacian can be formulated in a number of ways, of which we describe two. The first uses the weight matrix, W, which is symmetric. We will provide some intuition as

to the physical role these edge weights play later. The degree of the i^{th} vertex, d_i, is defined as the sum of all neighbouring edge weights

$$d_i = \sum_{i \sim j} w_{ij} \quad \forall \quad e_{ij} \in E \tag{2.1}$$

The un-normalized weighted graph Laplacian (WGL) of Γ is then given by (Chung, 1997) (see Appendix II A.8 for *diag* operator).

$$L = diag(d) - W \tag{2.2}$$

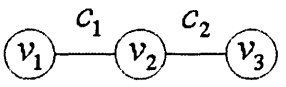
This is the discrete analogue of the LBO used to represent a diffusion process on a Riemannian manifold (Rosenberg, 1997; Sochen et al., 1998). We will discuss the analogous relationship between this operator and its discrete counterpart later in this chapter.

An alternative perspective is to construct the WGL using the edge-node incidence and constitutive matrices (Strang, 2004; Strang, 2007). The first of these, A, has dimensions $N_E \times N_V$ and has components (rows and columns are indexed by elements of the edge and vertex set respectively)

$$A_{e_{ij}v_k} = \begin{cases} +1 & if \quad i = k \\ -1 & if \quad j = k \end{cases} \tag{2.3}$$

This is the discrete analogue of the gradient operator and its transpose, A^T, the analogue of the divergence operator (Strang, 2007). We will use this to provide some intuition into the analogous relation between the WGL and LBO later. The second matrix, C, has dimensions $N_E \times N_E$ and is diagonal, containing edge weights, *e.g.* for the a^{th} edge, $C_{aa} = w_{ij}$. Given these, the WGL is given by

$$L = A^T C A \tag{2.4}$$



$$c_1 = w_{12} = w_{21} \qquad c_2 = w_{23} = w_{32}$$

Figure 2-1: 1D graph comprised of three nodes and two edges

Edge weights are symmetric and have been re-labelled, c_1 and c_2

A simple example of a 1D graph comprised of three nodes and two edges, configured as a chain, is shown in Figure 1. As the edge weights are symmetric we have re-labelled them as c_1 and c_2. The weight matrix is then

$$W = \begin{pmatrix} 0 & c_1 & 0 \\ c_1 & 0 & c_2 \\ 0 & c_2 & 0 \end{pmatrix} \qquad 2.5$$

using Eqn 2.1 the vector containing the degree of each node is

$$d = [c_1, c_1 + c_2, c_2]^T \qquad 2.6$$

and Eqn 2.2 gives the WGL

$$L = \begin{pmatrix} c_1 & -c_1 & 0 \\ -c_1 & c_1 + c_2 & -c_2 \\ 0 & -c_2 & c_2 \end{pmatrix} \qquad 2.7$$

Similarly, using Eqn 2.3 the incidence and constitutive matrices are

$$A = \begin{pmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix}$$

$$C = \begin{pmatrix} c_1 & 0 \\ 0 & c_2 \end{pmatrix} \qquad 2.8$$

and using Eqn 2.4 produces the same WGL as in Eqn 2.7.

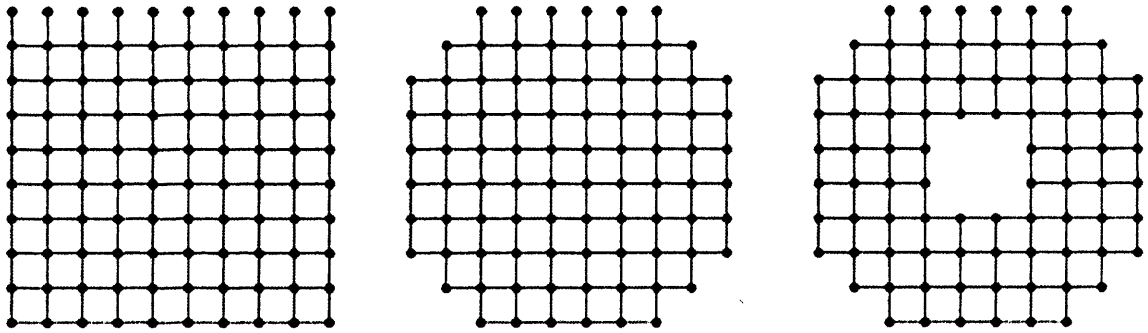


Figure 2-2: Regular (left) 2D graph and two with irregular boundaries

Nodes and edges are represented by dots and lines connecting them respectively. All three graphs have 4-nearest neighbor topology

This matrix appears in many different physical contexts, such as mass-spring systems where it is called the "stiffness" matrix (Strang, 2007, Hatch, 2000 #715), or electrical networks, where components along edges can be resistors, capacitors or inductors (Bamberg and Shlomo, 1990). Using the mass-spring analogy the graph in Figure 2-1 corresponds to three unit masses coupled by two springs, characterized by Hooke's constant, which corresponds to edge weights. Applying a force to the masses, ρ, results in their displacement, f, which at steady state is given by solving the equation

$$Lf = \rho \qquad 2.9$$

Looking at the Laplacian matrix in Eqn 2.7 we see that the all ones vector is an eigenvector of this system, which has eigenvalue equal to zero. This means that the matrix is singular, which is due to the boundary conditions implicit in this formulation of the stiffness matrix, which is that neither of the two masses at the end of the chain are fixed. Intuitively this eigenvector corresponds to rigid motion of the masses.

The WGL can be used to represent a 2D image using a 2D, or planar, graph, where pixel values are associated with the node set. For example, a grey-scale image would have one number, *i.e.* a 1×1 vector at each node, while a colour image would have a 3×1 vector at each node corresponding to red, green and blue channel intensities. In general there can be P 'channels' and so we represent image intensities on the node set by a $P \times N_V$ matrix, f (see later). Three examples of 2D graphs are shown in Figure 2-2. The graph on the left is

a product of two one dimensional graphs and is called a Cartesian or regular graph, whereas the two on the right are irregular in the sense that their boundaries prevent such a product space representation. The benefit of regular graphs is that they are computationally easier to handle. However, a volume of fMRI data contains many non-brain voxels and is the reason for choosing graphs with irregular boundaries to represent only regions we wish to analyse. An additional benefit of using irregular graphs is that the approach proposed in this thesis can also be used with regular graphs. As such we will use graphs with irregular boundaries and, when required, that exclude some interior nodes (see right most graph in Figure 2-2), for example, regions containing cerebral spinal fluid (see Figure 2 in Appendix I).

The edge weights of a graph play a crucial role, as intuited from the mass-spring analogy (as they define interactions between masses), which we consider next.

2.2 Edge weights of a graph

In this thesis we take each edge weight to be a function of a squared distance, $ds(v_i,v_j)^2 \geq 0$, between vertices, given by

$$w_{ij} = \begin{cases} \exp(-ds(v_i,v_j)^2) & for \quad i \sim j \\ 0 & otherwise \end{cases} \qquad 2.10$$

The distance, $ds(v_i,v_j)$ between vertices v_i and v_j, in general contains two components, $d\varepsilon$ and dg. Note that first of these is meant to symbolise Euclidean, i.e. not an error term as used in a GLM. These are distances in physical and feature space respectively, where the 'feature' at a vertex is the vector of image values (see matrix f below). The squared distance of the a^{th} edge connecting vertices v_i and v_j is given by

$$\begin{aligned} ds^2 &= d\varepsilon^2 + dg^2 \\ d\varepsilon^2 &= du_a^T H_d du_a \\ dg^2 &= df_a^T H_f df_a \end{aligned} \qquad 2.11$$

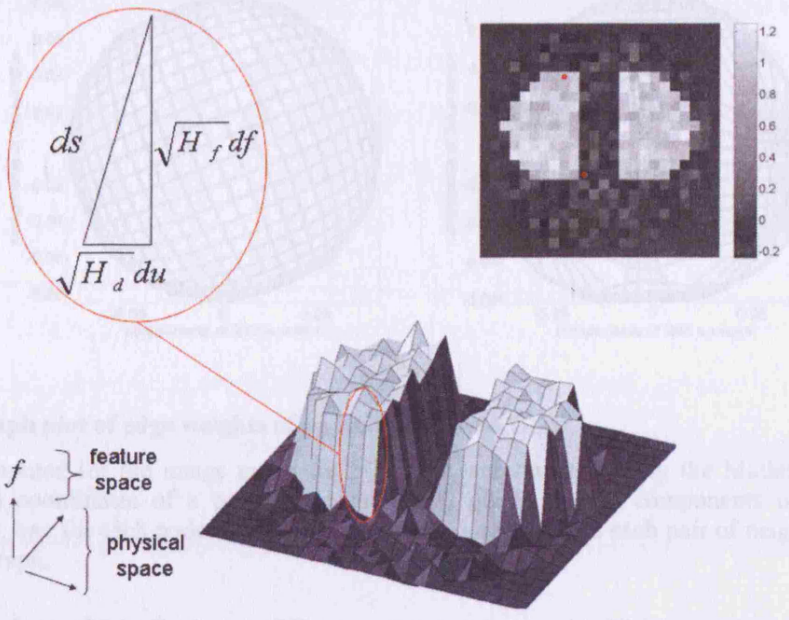


Figure 2-3: Image as a function over a graph

A 2D scalar image (top right) where pixel values are indicated by the greyscale. This can be represented in 3D where the first two dimensions are physical space and the third is a feature, i.e. pixel value (lower figure). The distance between adjacent points in this 3D representation depends on distance in physical and feature space, du and df (top left).

where $du_a = (du_a^1, du_a^2)^T$ is displacement in physical space and $df_a^T = A_a f^T$ is displacement in feature space, where f is a matrix containing image intensities (*e.g.* pixel values in an image or GLM parameter values) of dimension $P \times N_V$, and A_a is the a^{th} row of A. As such, the penultimate and last lines of Eqn 2.11 are squared distance in physical space and feature space respectively. The quantities, H_d and H_f scale the respective displacements and, in this thesis, we chose these to be the identity, *i.e.* $H_d = I_{n_d}$, where n_d is the number of spatial dimensions and fix H_f (see Appendix II J.1). If $H_f = 0$ then the Laplacian is based on Euclidean distance in physical space only, which results in an isotropic (*i.e.* no preferred direction) Laplacian matrix, otherwise the weights are anisotropic. Note that the spatial distances are not restricted to be Euclidean, e.g. they could be on a cortical mesh, in which case H_d would be non-trivial and vary with position on the mesh.

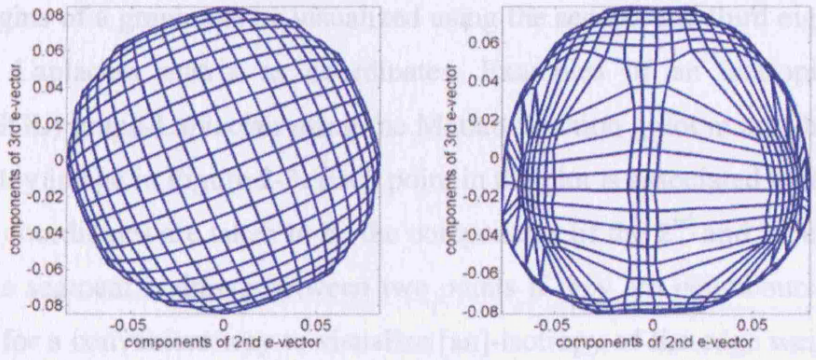


Figure 2-4: Graph plot of edge weights of an EGL (left) and GGL

These were computed for the image in Figure 2-3. Plots were created using the Matlab function gplot.m, where the (x,y) coordinates of a point, associated to a pixel, are the components of the 2nd and 3rd eigenvectors, i.e. one for each node, and a line segment is draw between each pair of neighbouring nodes, i.e an edge of the graph.

We illustrate these ideas using a 2D toy image shown in Figure 2-3, were $n_d = 2$ and $P = 1$, i.e. a scalar image. We will see examples of $n_d = 3$ and $P > 1$ for GLM parameter images over a brain volume in Chapter 4. The scalar image shown at the top right of Figure 2-3 is on an irregular graph, in that the boundary is circular, *i.e.* there are no nodes between this boundary and the outer rectangle. The image contains two regions of high pixel values compared to the background and Gaussian noise has been added. This image can be considered as a three dimensional object (lower figure), were pixel values at each node make the 3rd dimension, *i.e.* feature space. Distance can then be defined in this 3D space (see Eqn 2.11) and used to compute edge weights (see Eqn 2.10). An illustration of these distances between two pixels along an edge of an image (note that this is different to the edge of a graph) in this 3D space is shown in the upper left, whose squared length is given by Eqn 2.11. It is easily seen that if H_f is zero then the distance is in physical space only, in which case, edge weights are not a function of pixel values. In this thesis we refer to this as a Euclidean graph-Laplacian (EGL), which is isotropic. If the scaling is non-zero then pixel values are encoded in edge weights, which we refer to as a geodesic graph Laplacian (GGL) as it depends on geodesic distance between features at neighbouring nodes. This is in general anisotropic. As such the EGL is a special case of the GGL.

The edge weights of a graph can be visualized using the second and third eigenvectors (see later) of the Laplacian matrix as coordinates. Examples of an isotropic (EGL) and anisotropic (GGL) graph-Laplacian using the Matlab function gplot.m are shown in Figure 2-4 using the toy image in Figure 2-3. Each point in the plot is associated with a node of the graph, whose coordinates are taken to be the components of the 2nd and 3rd eigenvectors of the GL. A line segment is drawn between two points if they are neighbours on the graph, which makes for a convenient way to visualize [an]-isotropy of the edge weights. Isotropic weights are easily seen on the left compared to the right, which reveals a representation of the three regions in the image of Figure 2-3, *i.e.* two regions with high pixel values and background, which are separated by large distances corresponding to steep edges of the image between high/low pixel values.

2.2.1 Relation to a continuous representation

Despite using a discrete formulation it is instructive to consider analogies with continuous schemes. Sochen *et al* considered an image as a surface embedded in a higher dimensional space (Aubert and Kornprobst, 2002; Kimmel, 2003; Sochen et al., 1997). This leads to using the LBO instead of the WGL and a PDE-based approach in place of diffusion on a graph. We will first describe a simple example, where we consider a scalar function, $f(u)$, over the domain, u, as a curve embedded in a 2D Euclidean space, as shown in Figure 2-5.

Looking at this figure it is easy to appreciate that a small distance on the domain, du, is related to the distance, ds, on the curve (two examples are shown in bold) by

$$\begin{aligned} ds^2 &= du^2 + df^2 \\ &= du^2\left(1+\left(\frac{df}{du}\right)^2\right) \end{aligned} \qquad 2.12$$

As such the quantity in brackets varies with position, u.

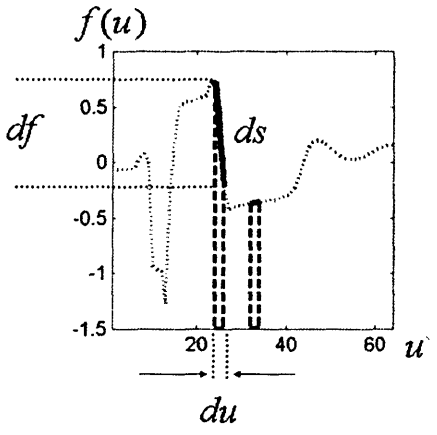


Figure 2-5: Embedding a 1D space, i.e. a curve, in two dimensions

The approximate gradient of the function, $f(u)$, is shown at two regions of u.

An alternative perspective is to consider the two spaces involved; the domain, u, and $(u, f(u))$ within which the curve is embedded, referred to as the embedding space or feature-space manifold (Sochen et al., 1997). We denote each space by M and N respectively. Each of these will have a metric tensor[10] that defines distance within the space, denoted by G and H. Given that we have chosen the embedding space to be Euclidean, the second of these is the identity matrix (for the all points in this space), while the first is, in general non-trivial, in that it depends on position, u, and is known as the induced metric tensor (more on this important quantity shortly). We then consider a map that goes from M to N, which we denote by χ

$$\begin{aligned} &\chi : M \to N \\ &\chi : u \to (\chi^1(u), \chi^2(u)) = (u, f(u)) \end{aligned} \qquad 2.13$$

The metric tensor G is *induced* in that it is specified in terms of the embedding space metric tensor, H, and the map, χ. This is computed using the Jacobian, J, of the map, which has components (where derivatives are represented as $f_x = \partial f / \partial x$)

$$J_{ij} = \frac{\partial \chi^i}{\partial u^j} \Rightarrow J = \begin{bmatrix} 1 & f_u \end{bmatrix}^T \qquad 2.14$$

[10] The metric tensor of a Riemannian manifold is used to define length, orthogonality and volume. The induced metric tensor is that of a sub-manifold. A manifold is a generalization of a space and a Riemannian space is one whose metric tensor is positive definite. See the footnotes of (Smith, 2005) for an informal description of key ideas in differential geometry.

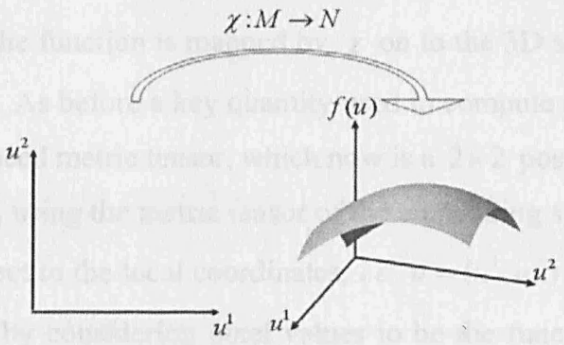


Figure 2-6: Embedding a 2D space, i.e. a surface, in three dimensions

The space (u^1, u^2) is mapped into the space $(u^1, u^2, f(u))$ by χ.

and the equation

$$G = J^T H J \Rightarrow G = 1 + f_u^2 \qquad 2.15$$

The squared length of a vector in the space M is then given by the inner product

$$\begin{aligned} ds^2 &= du^T G du \\ &= du^2 (1 + f_u^2) \end{aligned} \qquad 2.16$$

As such we can represent the curve embedded in 2D using a 1D space that has a non-trivial (induced) metric tensor G, which varies with position in u. Comparing Eqns 2.16 and 2.12 we see that $G \approx 1 + (df/du)^2$, where the right hand side is an approximation based on finite differences. An important quantity, which we shall see again later is $\sqrt{\det(G)}$, which in this case is trivial as G is a scalar, because it measures the ratio of distances on the domain and curve, i.e. $\sqrt{\det(G)} = ds/du$. This quantifies the amount a unit length (in Euclidean space) is scaled (or magnified) on the curve.

Extending this to a 2D surface embedded in a 3D space, we consider the domain $u = (u^1, u^2)$. The scalar function, $f(u)$, can then be considered as a surface embedded in a 3D Euclidean space if we use the map

$$\begin{aligned} &\chi : M \to N \\ &\chi : u \to (\chi^1(u), \chi^2(u), \chi^3(u)) = (u^1, u^2, f(u)) \end{aligned} \qquad 2.17$$

i.e. the 2D surface of the function is mapped by χ on to the 3D space. A schematic of this is shown in Figure 2-6. As before a key quantity used to compute distance on the surface of the function is the induced metric tensor, which now is a 2×2 positive definite matrix. This is computed, as before, using the metric tensor of the embedding space, H, and Jacobian of the map, J, with respect to the local coordinates, *i.e.* $u=(u^1,u^2)$. These ideas can be used to represent an image by considering pixel values to be the function, $f(u)$, where u are coordinates in physical space. Before we consider our toy image in light of these ideas, we will illustrate them further by considering a familiar 2D surface embedded in a 3D space; the surface of a sphere (Sochen et al., 1997), which is shown in Figure 2-7.

The local coordinates are given by the latitude, ϕ, (angle from the "north pole") and longitude, θ, (angle in the $x-y$ plane); $u=(\phi,\theta)$. The map from this two dimensional Riemannian surface to $\Re^3$ is

$$\begin{aligned}&\chi: M\to N\\&\chi:(\phi,\theta)\to(\sin\phi\cos\theta,\sin\phi\sin\theta,\cos\phi)\end{aligned} \qquad 2.18$$

The Jacobian of the map is

$$J=\begin{pmatrix}\cos\phi\cos\theta & -\sin\phi\sin\theta\\ \cos\phi\sin\theta & \sin\phi\cos\theta\\ -\sin\phi & 0\end{pmatrix} \qquad 2.19$$

and so the induced metric tensor is

$$\begin{aligned}G&=\begin{pmatrix}\cos\phi\cos\theta & \cos\phi\sin\theta & -\sin\phi\\ -\sin\phi\sin\theta & \sin\phi\cos\theta & 0\end{pmatrix}\begin{pmatrix}\cos\phi\cos\theta & -\sin\phi\sin\theta\\ \cos\phi\sin\theta & \sin\phi\cos\theta\\ -\sin\theta & 0\end{pmatrix}\\&=\begin{pmatrix}1 & 0\\ 0 & \sin^2\phi\end{pmatrix}\end{aligned} \qquad 2.20$$

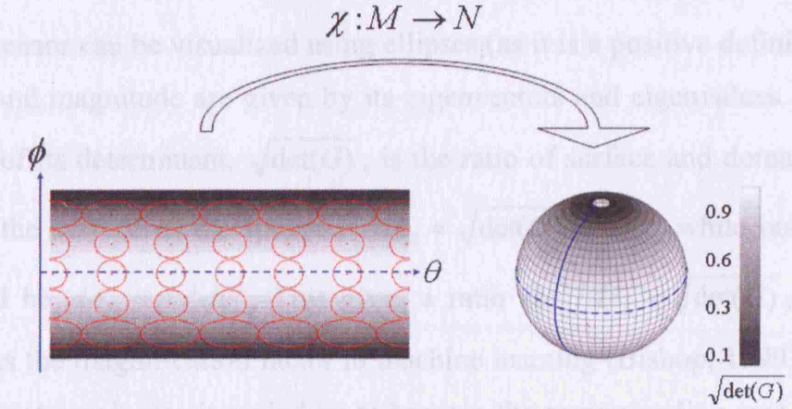


Figure 2-7: Surface geometry of a sphere
On the right a sphere is shown with coordinates; latitude and longitude (solid and dashed curves). A representation of the surface of the sphere is shown on the left. The metric tensor of this space varies with position, which is indicated by the red ellipses that represent the 2×2 tensor G. The square root of the determinant of this metric represent the amount a unit of area is scaled going from a flat space to the surface of the sphere. Note that this approaches zero at the poles.

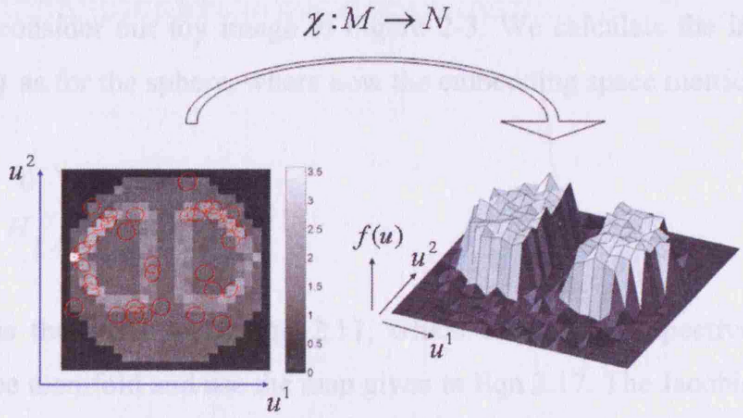


Figure 2-8: Induced metric tensor of a scalar image
The 2D image in Figure 2-3 is considered as a surface embedded in 3D space. The induced metric tensor and the square root of its determinant are shown as in Figure 2-7. Of note is the increase in area and alignment of G with the edges of the image. This leads to diffusion that is predominantly along an edge of the image as opposed to across it.

Where we have used the trigonometric identity, $\cos^2 x + \sin^2 x = 1$ and embedded the surface in a Euclidean space, i.e. $H = I_3$. Distance on the surface of the sphere is then given by

$$\begin{aligned} ds^2 &= du^T G du \\ &= \begin{pmatrix} d\phi & d\theta \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & \sin^2 \phi \end{pmatrix} \begin{pmatrix} d\phi \\ d\theta \end{pmatrix} \\ &= (d\phi)^2 + (\sin\phi d\theta)^2 \end{aligned} \qquad 2.21$$

The metric tensor can be visualized using ellipses (as it is a positive definite matrix), where orientation and magnitude are given by its eigenvectors and eigenvalues respectively. The square-root of its determinant, $\sqrt{\det(G)}$, is the ratio of surface and domain areas. An area element on the surface of the sphere is $dA_S = \sqrt{\det(G)} du^1 du^2$, while one on a Euclidean space would be $dA_D = du^1 du^2$. This gives a ratio $dA_S / dA_D = \sqrt{\det(G)}$, which has been referred to as the magnification factor in machine learning (Bishop, 1999). In other words, it is the amount a unit area is scaled by to become the corresponding area on the surface of the sphere. This is a scalar quantity that can be calculated at each location on the sphere. This is shown along with ellipses representing the matrix G at a number of positions in Figure 2-7, where we notice that G and $\sqrt{\det(G)}$ depend on position.

We now reconsider our toy image in Figure 2-3. We calculate the induced metric tensor, G, on $f(u)$ as for the sphere, where now the embedding space metric has two components

$$H = \begin{pmatrix} H_d & 0 \\ 0 & H_f \end{pmatrix} \qquad 2.22$$

where H is the same as in Eqn 2.11, which scales the respective components of the feature-space manifold and use the map given in Eqn 2.17. The Jacobian of this is

$$J = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ f_{u^1} & f_{u^2} \end{pmatrix} \qquad 2.23$$

The induced metric tensor is then

$$\begin{aligned} G &= J^T H J \\ &= H_d + f_u^T H_f f_u \end{aligned} \qquad 2.24$$

which is used to calculate distance as before. Note that if $f_u \approx df/du$, where du and df are displacements in physical and feature space as in Eqn 2.11, then substituting Eqn 2.24 into the top line of 2.16, produces the approximation

$$\begin{aligned} ds^2 &= du^T\left(H_d + f_u^T H_f f_u\right)du \\ &\approx du^T H_d du + df^T H_f df \end{aligned} \qquad 2.25$$

which agrees with the squared distance in Eqn 2.11.

The magnification factor and ellipses representing G from a random selection of points in the image are shown in Figure 2-8 for our toy image as for the sphere. Flat regions have values of det(G) of about one, while edges are greater than unity. High values correspond to locations where the distance on $f(u)$ between adjacent pixels (see Figure 2-3) is large; *i.e.*, at an edge of the image where gradients are large. It can be seen that the metric tensor is aligned with the edges of the two central regions of high pixel values and isotropic elsewhere. This means that the anisotropic and non-stationary nature of the image is encoded in G.

The benefits of this approach are that non-trivial domains can be used, such as the surface of a sphere or cortical surface, and it is easily extended to vector valued images. An example of the latter on a flat surface is a colour image comprised of three channels. The map in this scenario is then

$$\begin{aligned} &\chi : M \to N \\ &\chi : u \to (\chi^1(u), \chi^2(u), \chi^3(u), \chi^4(u), \chi^5(u)) = (u^1, u^2, f^1(u), f^2(u), f^3(u)) \end{aligned} \qquad 2.26$$

where now $f(u)$ is a vector-valued function, in this case producing a 3-vector at each point in physical space. These ideas can be used to generalize the Laplace operator to curved surfaces, i.e. the Laplace-Beltrami operator (LBO), which we consider next.

The aim here is to provide some intuition by looking at similarities between the LBO and the formula for the WGL in Eqn 2.4. In component form the LBO is written

$$\Delta_G = \frac{1}{\sqrt{\det(G)}} \partial_i \sqrt{\det(G)} G^{ij} \partial_j \qquad 2.27$$

where G^{ij} represents components of the inverse, G^{-1}, $\partial_i = \partial / \partial u^i$ and we have used the subscript, Δ_G, to distinguish it from the Laplace operator, Δ, which is defined on a Euclidean space, i.e. where G is equal to the identity at all points in space.

To simplify the analogy we return to the 1D example earlier (see Figure 2-5). Focusing on the main part of Eqn 2.27, we can write this in a suggestive form

$$\begin{aligned} \Delta_G &\propto div(c(f)grad) \\ c(f) &= \sqrt{\det(G)} G^{-1} \end{aligned} \qquad 2.28$$

Where G was given in Eqn 2.15 and depends on the gradient, f_u. The top line is suggestive in that it has the same form as Eqn 2.4 (shown below for convenience), in that it has two operations either side, what can be thought of as, a conductivity function. The discrete analogues of the gradient and divergence operators are the edge-node incidence matrix and its transpose. The analogue of the conductivity function, $c(f)$, is the constitutive matrix and the WGL is given by

$$L = A^T C(f) A \qquad 2.29$$

This provides some insight into the analogous relationship between the LBO and WGL, which is useful in that previous work from both perspectives can inform the development of adaptive spatial priors. More details on the correspondence between the WGL and LBO, in the contexts of nonlinear dimensionality reduction and clustering, can be found in (Belkin and Niyogi, 2003).

2.3 Eigensystem of a graph Laplacian

The Laplacian matrix can be represented by its eigensystem (Strang, 2004), which is a decomposition into eigenvectors and eigenvalues, where the i^{th} eigenvalue and vector are

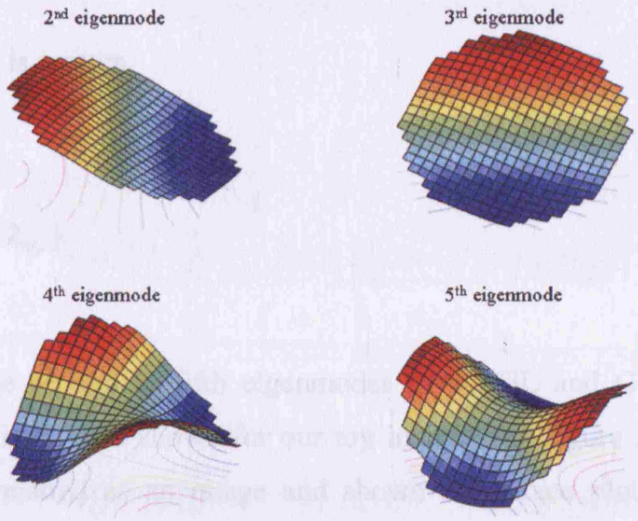


Figure 2-9: Eigenmodes of an isotropic graph-Laplacian (EGL)

Surface plots of the first four non-trivial eigenvectors (reformatted as images) are shown for the toy image in Figure 2-3

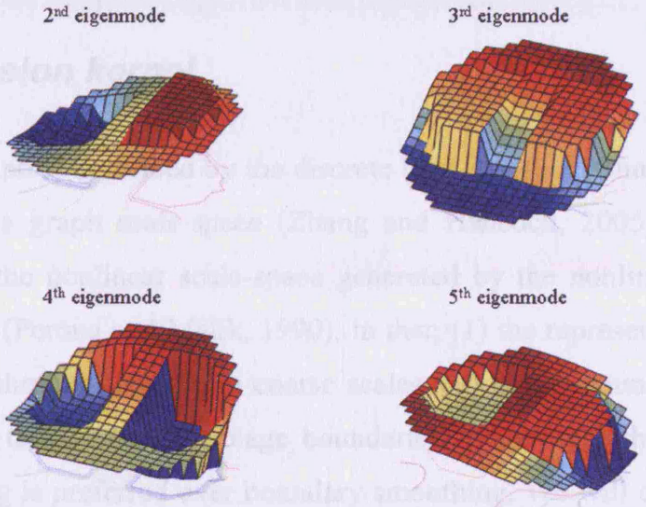


Figure 2-10: Eigenmodes of an anisotropic graph-Laplacian (GGL)

represented by λ_i and ϕ_i (a column vector of length N_V) respectively. As all rows sum to zero, i.e. the all ones vector is an eigenvector with eigenvalue zero, it is positive semi-definite, i.e. $\lambda_i \geq 0$. The Laplacian is then a weighted sum of the outer product of eigenvectors, i.e. $\phi_i \phi_i^T \in \Re^{N_V \times N_V}$, given by

$$L = \sum_{i=1}^{N_V} \lambda_i \phi_i \phi_i^T \qquad 2.30$$

In matrix form this is written

$$L = \Phi \Lambda \Phi^T$$

$$\Lambda = diag(\lambda_1, \lambda_2, ..., \lambda_{N_V}) \qquad 2.31$$
$$\Phi = \left[\phi_1, \phi_2, ..., \phi_{N_V}\right]$$

For illustration, the second to fifth eigenmodes of a EGL and GGL (*i.e.* isotropic and anisotropic respectively) are shown for our toy image (see Figure 2-3) in Figure 2-9 and Figure 2-10 (reformatted as an image and shown as surface plots). Note that the first eigenmode is not included as this is constant over the graph. These eigenmodes provide a basis set over nodes on the graphs, that is; vectors of length N_V can be represented using linear combinations of these eigenmodes.

2.4 The diffusion kernel

Diffusion on a graph is described by the discrete heat equation (Chung, 1997). This can be used to generate a graph scale-space (Zhang and Hancock, 2005) representation of an image similar to the nonlinear scale-space generated by the nonlinear PDE proposed by Perona and Malik (Perona and Malik, 1990), in that; (1) the representation has no spurious details generated through the fine to coarse scales, (2) object boundaries remain sharp at coarse scales and the location of image boundaries do not shift through scale-space, (3) regional smoothing is preferred over boundary smoothing. We will consider a constant GL here, however, in general diffusion of an image changes its pixel values, which in turn changes the GGL. This means that ideally the GGL should be recomputed after each iteration of diffusion. However, in our experience compelling results are obtained using a GGL based on the initial image (in the context of time-series this translates to the OLS estimate using non-smoothed data). We review this approximation in the Discussion and provide details for updating the GGL in Appendix II K.

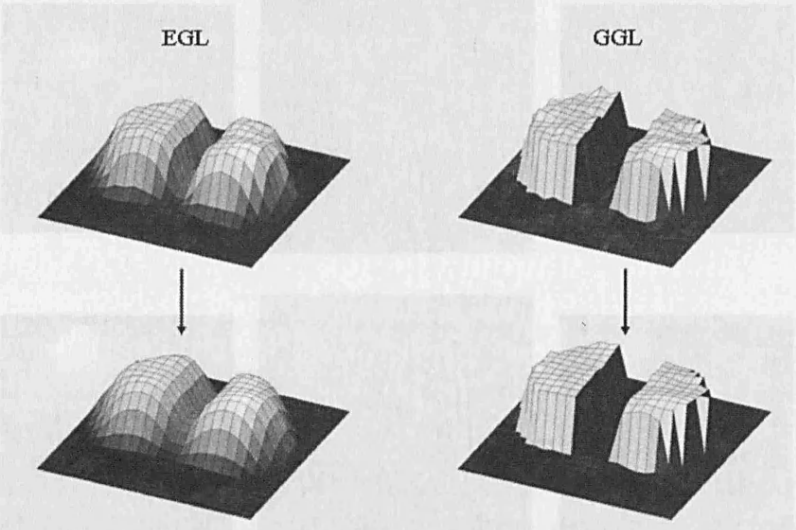


Figure 2-11: Solutions to the diffusion equation

The initial condition was the image in Figure 2-3. Diffusion using the EGL kernel (left) acts as a low pass filter, removing high spatial frequencies, which includes edges of the original image. This is not so for diffusion using the GGL kernel that removes high frequency noise, but not at the expense of edges of the image.

Given the steady state equation $\rho = Lf$ (see Eqn 2.9), where $f(t) \in \Re^{N_V \times 1}$, we can consider this in a dynamic context by taking $\rho = -2\,df/dt$ (Strang, 2007), to recover the diffusion equation

$$\frac{df}{dt} = -\tfrac{1}{2} Lf \qquad 2.32$$

which has the solution

$$\begin{aligned} f(t+dt) &= P_{dt} f(t) \\ P_{dt} &= \exp(-\tfrac{1}{2} Ldt) \end{aligned} \qquad 2.33$$

Where the matrix, P_{dt}, is the matrix exponential of the Laplacian matrix. This is the local solution to the heat equation on a graph, which propagates the function, $f(t)$, on nodes of the graph, from time t to $t+dt$.

We show the effect of diffusion at two different times, in Figure 2-11, using our toy image as the initial condition. Surface plots of the image are snap shots of the graph scale-space (Zhang and Hancock, 2005) representation generated by the discrete diffusion equation.

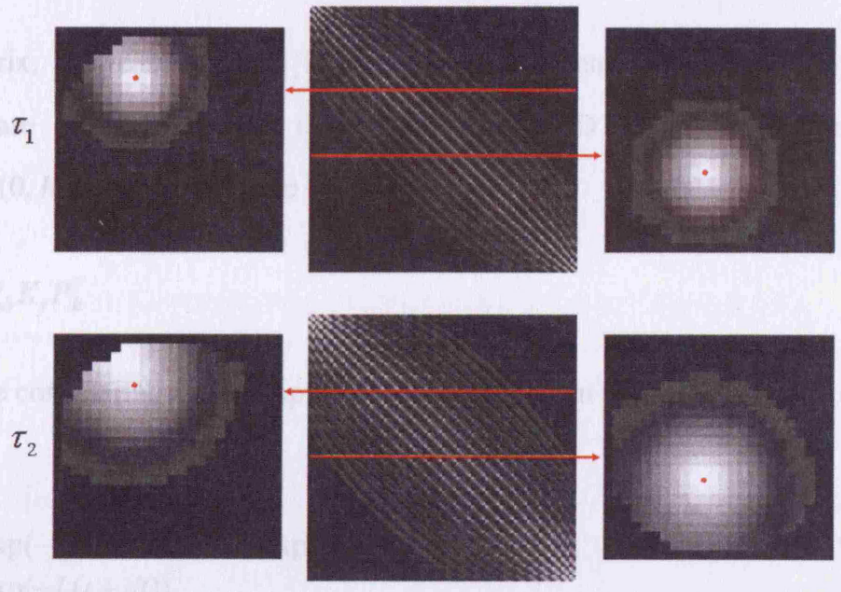


Figure 2-12: Diffusion kernels of an isotropic graph-Laplacian (EGL)
(central column) Matrices representing the diffusion kernel show increased dispersion after diffusion for longer ($\tau_1 < \tau_2$). Local kernels centred at two locations are shown either side, which are rows of the diffusion kernel (indicated by the red lines) reformatted as an image. These reveal the isotropy of the kernel.

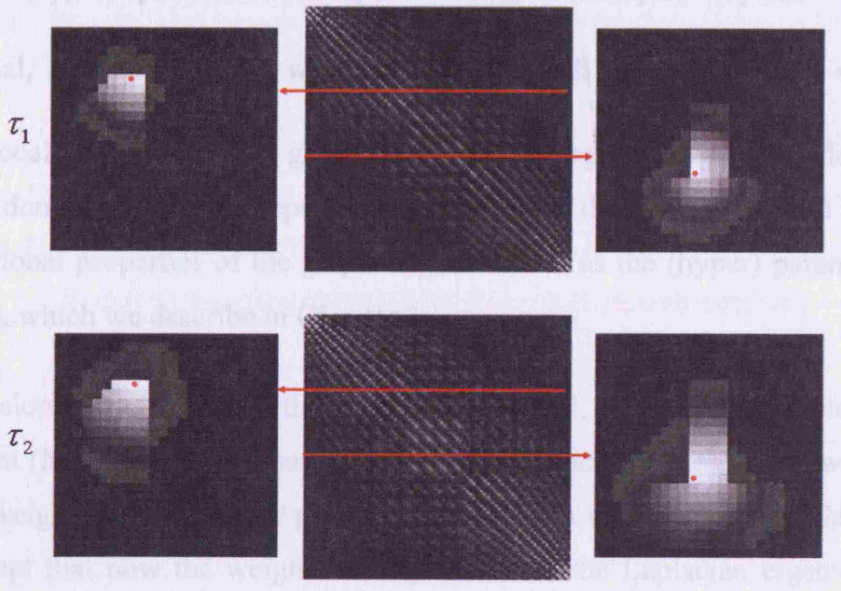


Figure 2-13: Diffusion kernels of an anisotropic graph-Laplacian (GGL)
Same layout as above. Anisotropy of the kernel is seen in both the covariance matrix (central column) and local kernels.

This shows smoothing of high spatial frequencies for diffusion using an EGL and preservation of edges of the image when using a GGL.

The matrix, P_{dt} can be used to represent a discrete random field by considering a multivariate normal density (see Appendix II D for notation) over $f(t)$, *i.e.* $f_t \sim N_{N_V}(0, K_t)$. The covariance at $t+dt$ is then

$$K_{t+dt} = P_{dt} K_t P_{dt}^T \qquad 2.34$$

As we are considering the example where the Laplacian matrix is constant over time, this is given by

$$\begin{aligned} K_{t+dt} &= \exp(-\tfrac{1}{2} L dt) \exp(-Lt) \exp(-\tfrac{1}{2} L dt) \\ &= \exp(-L(t+dt)) \end{aligned} \qquad 2.35$$

The Gaussian density at $\tau = t+dt$ over $f(\tau)$ is then $f_\tau \sim N_{N_V}(0, K_\tau)$, i.e. the covariance matrix is (hyper) parameterized by τ. Using the Taylor expansion of the matrix exponential, $\exp(A) = \sum_{n=1}^{\infty} \frac{A^n}{n!}$, we note that for small values of τ, $K \approx I - L\tau$, which encodes local properties of the graph. However, at larger values of τ, global structure of the graph dominates. That is, depending on the time of diffusion the kernel represents either local or global properties of the graph. This is useful as the (hyper) parameter, τ, can be optimized, which we describe in Chapter 3.

The diffusion kernel involves the matrix exponential, which can be computed once the eigensytem (Moler and Van Loan, 2003) of the Laplacian matrix is known. The kernel is then the weighted sum of outer products of Laplacian eigenvectors, similar to that in Eqn 2.31, except that now the weights are a *function* of the Laplacian eigenvalues, which in general we denote by $g(\lambda_i)$ and in particular is $g(\lambda_i) = \exp(-\lambda_i \tau)$ for the diffusion kernel. These are the eigenvalues of the diffusion kernel, *i.e.* it is a function of the eigensystem of the Laplacian matrix.

$$K = \sum_{i=1}^{N_V} g(\lambda_i, \tau) \phi_i \phi_i^T \qquad 2.36$$

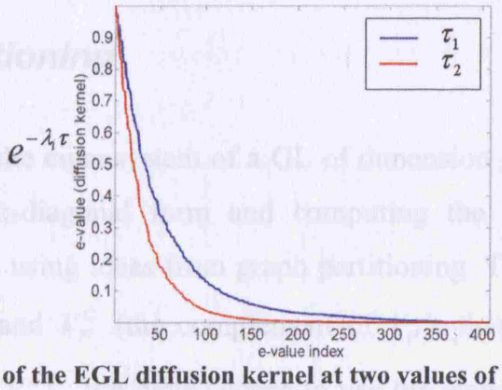


Figure 2-14: Eigenvalues of the EGL diffusion kernel at two values of τ
Note the rapid decay of values, in particular, as τ increases ($\tau_1 < \tau_2$)

In matrix notation this is

$$K = \Phi g(\Lambda)\Phi^T$$
$$g(\Lambda) = \exp(-\Lambda\tau) \qquad 2.37$$

Note that all eigenvalues of the diffusion kernel are greater than zero, due to the exponential, that is; it is a positive definite matrix. Examples of this matrix are shown in Figure 2-12 and Figure 2-13, for a EGL and GGL at two different values of τ, where $\tau_1 < \tau_2$. A more intuitive way to see the difference in dispersion for the isotropic and anisotropic kernels is to look at a row of the kernel reformatted as an image, which we refer to as a local kernel. This contains the weights used to average a spatial signal and are shown for two points in the image either side of the diffusion kernel.

A plot of eigenvalues of the isotropic diffusion kernel are shown in Figure 2-14, again at two different values of τ. Note the increased rate of decay of weights (diffusion kernel eigenvalues) for larger values of τ. Higher order eigenmodes contain high spatial frequency components, which means that as τ increases the diffusion process down weights these high frequency modes, which has the effect of smoothing a function over the graph, *i.e.* variability in the function is reduced with time of diffusion, e.g. see the effect of diffusion using the EGL kernel in Figure 2-11 (left).

We will use these properties of the diffusion kernel to define a spatial covariance matrix of a spatial prior in Chapter 3.

2.5 Graph Partitioning

Instead of computing the eigensystem of a GL of dimension $\sim 10^{5-6}$, we approximate it by dividing it into block-diagonal form and computing the eigensystem for each block separately. We do this using ideas from graph partitioning. The task is to separate a graph into two subsets, V_P and V_P^C (the complement of V_P), that share a minimal number of edges, where $V = \{V_P, V_P^C\}$. We have chosen to use the isoperimetric algorithm instead of spectral techniques as it involves solving a system of equations instead of an eigenvalue problem, which improves speed and numerical stability (Grady and Schwartz, 2006). An additional benefit is that it has the potential to be used in a Mixture of Experts (Bishop and Svensen, 2003) model, which has soft instead of hard partition boundaries (see Discussion). We provide a brief outline of the algorithm here (see Grady and Schwartz, 2006 for further details) and present a summary of the steps used in our implementation.

The isoperimetric problem is; *for a fixed area, find the shape with minimal perimeter* (Chung, 1997). The isoperimetric number (Mohar, 1989), h_G, is the infimum of the ratio of the area of the boundary and the volume of V_P, where the boundary, $|\partial V_P|$, is defined as $\partial V_P = \{e_{ij} \mid i \in V_P, j \in V_P^C\}$. The subset, V_P, can be defined by a binary indicator vector, x

$$x_i = \begin{cases} 0 & if \quad v_i \in V_P \\ 1 & if \quad v_i \in V_P^C \end{cases} \qquad 2.38$$

In which case, the boundary and volume of V_P are given by $|\partial V_P| = x^T L x$ and $vol_{V_P} = x^T d$. The isoperimetric number is then the *minimum* of their ratio

$$h_G = \min_x \left\{ \frac{x^T L x}{x^T d} \right\} \qquad 2.39$$

This minimization can be cast as the solution of a system of linear equations by allowing x to take non-negative real values (instead of binary) and solving Eqn 2.39 using Lagrange

multipliers. This results in the equation, $Lx = d$, which can be solved by specifying a boundary condition to remove the singularity in L. The boundary condition can be thought of, in terms of the electrical circuit analogue of a graph (Strang, 2004), as equivalent to selecting a ground node (vertex). This provides the boundary condition required to render L non-singular and thereby invertible. This additional constraint amounts to removing the row and column of the ground node from the full Laplacian matrix, L, to give the reduced Laplacian, L_0. We reduce both other quantities to get

$$L_0 x_0 = d_0 \qquad 2.40$$

which can be solved easily for $N_V \sim 10^5$, using standard Matlab routines or relaxation methods (Press et al., 2007) for $N_V > 10^5$. In the circuit analogue, the solution, x_0, is the measurement of potential at all other nodes in the circuit. This provides a function over the graph, which monotonically increases from the ground node. The vector x_0 is converted into binary form by specifying a threshold, t, *i.e.*, $V_P = \{v_i \mid x_i \leq t\}$ and $V_P^C = \{v_i \mid x_i > t\}$. This partition is referred to as a *cut*. Each segment can then be divided again[11], resulting in a recursive partitioning algorithm.

The ground node can be selected in a number of ways; either the node with maximal degree or at random. We have chosen the latter of these in our implementation, the steps of which are provided below:

1. Define a vertex set containing the volume to be analysed

2. Compute the graph-Laplacian for the whole graph

3. Select a ground node (at random)

4. Solve Eqn 2.40

[11] The algorithm is guaranteed to return a connected sub-graph for V_P, however, this is not so for V_P^C, *i.e.* V_P^C could contain more than one region.

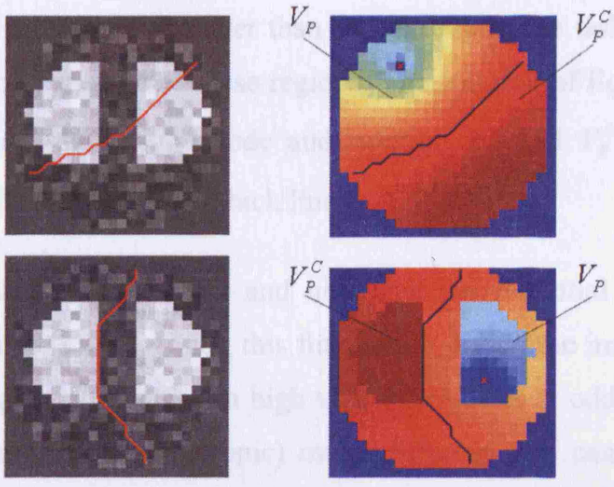


Figure 2-15: Partitioning an image using a graph-Laplacian

The nodes from the image in Figure 2-3 has been divided into two non-overlapping pieces, i.e. the subsets V_P and V_P^C, using the EGL (top row) and GGL. The first column shows the image to be partitioned along with the partition boundary (red line) computed. The second row shows the potential function emanating from the ground node, where it is zero, on which the partition is based.

5. Partition the vertex set into V_P and V_P^C

6. Ensure that V_P^C is connected (if not, return to 3)

7. Separate the Laplacian for each segment and go to 3

8. Stop if the number of vertices of a sub-graph is below a threshold

This algorithm ensures that each segment is connected and contains a similar number of voxels. Note that segments are locally connected as we assume a GL with nearest neighbour connectivity. This assumption could be relaxed by including long-range connections to couple non-local regions. An example of this in image processing is the use of small-world networks (Grady and Schwartz, 2004).

We demonstrate the algorithm for volumes of time series data in Chapter 4 and provide an illustration here using our toy example, shown in Figure 2-15. This shows a partition of the image using a EGL (upper row) and GGL (lower row), *i.e.* where $dg = 0$ and $dg \neq 0$

respectively (see Eqn 2.11). The image to be partitioned is shown in the first column, which contains two regions of pixel values greater than the surround. The task is to partition the image predominantly along an edge of these regions. The solution of Eqn 2.40 is shown in the second column, where the ground node and subsets, V_P and V_P^C, are indicated. A partition based on this is indicated by the thick line.

We consider first the EGL (upper row) and note that the potential function does not represent pixel values in the image. Using this function to divide the image produces a cut that passes through a region of pixels with high values, which is at odds with the task. As there is no preferred direction (*i.e.* isotropic) many different cuts can be selected given different ground nodes, the majority of which will not achieve the task. Compare this to the partition achieved using the GGL (lower row). First, we notice that the partition is predominantly along one of the steepest edges of the image. We see why this is so by examining the potential function, which now encodes pixel values as well (because $dg \neq 0$). Given this function, a cut is selected, which results in a partition that respects image boundaries.

In the next chapter, we will use the material covered in this chapter to formulate diffusion-based spatial priors for fMRI data.

3 Diffusion-based spatial priors for fMRI

In this chapter we use the material covered so far to formulate a spatial model of fMRI data. In particular, we describe how the diffusion kernel of a weighted graph Lapalcian can be used to encode spatial correlations between GLM parameters of fMRI time-series data. A number of simplifying assumptions are required for an efficient implementation, which we collate at the end of this chapter and consider in the discussion.

3.1 The model

In this section, we formulate a two-level GLM in terms of matrix-variate normal (MVN) densities (Gupta and Nagar, 2000). Our focus is the formulation of a multivariate normal model, with emphasis on covariance components and their hyper-parameters. We start with a linear model, under Gaussian assumptions, of the form

$$\begin{array}{ll} Y = X\beta + \varepsilon_1 & p(Y,\beta \mid X) = p(Y \mid X,\beta)p(\beta) \\ \beta = \varepsilon_2 & \Rightarrow \quad p(Y \mid X,\beta) = N_{r_1,c_1}(X\beta, S_1 \otimes K_1) \\ \varepsilon_i \sim N_{r_i,c_i}(0, S_i \otimes K_i) & p(\beta) = N_{r_2,c_2}(0, S_2 \otimes K_2) \end{array} \qquad 3.1$$

where the subscripts 1 and 2 indicate first and second levels in the hierarchy, i.e. data and parameter images respectively. The left-hand expressions specify a hierarchical linear model and the right-hand defines the implicit generative density in terms of a likelihood, $p(Y \mid X,\beta)$ and prior, $p(\beta)$. $N_{r,c}$ stands for a MVN density, where the matrix $A \in \Re^{r \times c}$, has probability density function (pdf), $p(A) \sim N_{r,c}(M, S \otimes K)$, with mean, M, of size $r \times c$, and two covariances, S and K, of size $r \times r$ and $c \times c$, for rows and columns respectively (see Appendix II D.3). Here, Y is a $T \times N_V$ data matrix, with T observations (i.e. scans) at each of N_V voxels, and X is a $T \times P$ design matrix, *i.e.* that contains P explanatory variables (columns of X) with an associated unknown $P \times N_V$ parameter matrix, β, so that $r_1 = T$, $r_2 = P$, $c_1 = c_2 = N_V$.

The first equation on the left of Eqn 3.1 is a typical model used in the analysis of fMRI data where the design matrix, X, contains explanatory variables of interest, e.g. stimulus onsets

convolved with a HRF (and possibly its derivatives) and variables of no interest e.g. a discrete cosine set to model scanner drift. As described in the introduction this models the response, Y, as a linear combination of columns in the design matrix plus a noise term. The aim is to estimate the posterior density over β, i.e. the parameter images of the GLM. In particular, we are interested in modelling spatial correlations of β. The errors at both levels are decomposed into rows and columns such that covariance S_i is over time or regressors and K_i over voxels. The addition of the second level places empirical shrinkage priors on the parameters.

This model can now be simplified by vectorising each component, i.e. reshaping all matrices in Eqn 3.1 to be column vectors, using the identity $vec(ABC) = (C^T \otimes A)\vec{B}$, where $\vec{B} = vec(B)$, i.e. the columns of B are stacked one on another progressively from the first to last. This leads to the model in terms of multivariate Gaussian densities

$$\begin{array}{ll} y = Zb + e_1 & p(y,b \mid Z) = p(y \mid Z,b)p(b) \\ b = e_2 \quad \Rightarrow & p(y \mid Z,b) = N_{n_1}(Zb, \Sigma_1) \\ e_i \sim N_{n_i}(0, \Sigma_i) & p(b) = N_{n_2}(0, \Sigma_2) \end{array} \qquad 3.2$$

Where $y = \vec{Y}$, $Z = I_{N_V} \otimes X$, $b = \vec{\beta}$, $e_i = \vec{\varepsilon}_i$, $n_i = c_i r_i$ and $\Sigma_i = K_i \otimes S_i$. $\otimes$ is the Kronecker product of two matrices, I_{N_V} is the identity matrix of size N_V and the unknown covariances of the first and second level errors, $\Sigma(\alpha)_1$ and $\Sigma(\alpha)_2$, depend on hyper-parameters, α. The model parameters and hyper-parameters are estimated using expectation maximization (**EM**) by maximising the log-marginal likelihood (see Appendix II E.8)

$$\begin{array}{l} F(q(b),\alpha) = -\frac{1}{2}\left(\ln|\Sigma| + y^T\Sigma^{-1}y + TN_V \ln 2\pi\right) \\ \Sigma(\alpha) = \Sigma_1 + Z\Sigma_2 Z^T \end{array} \qquad 3.3$$

with respect to the posterior density over parameters, which we denote by $q(b)$, in the **E**-step and the covariance hyper-parameters, i.e. α, in the **M**-step (model inversion with **EM** will be described later). Here, $\Sigma(\alpha)$ represents the covariance of the data induced by both

levels of the model. There are several ways of writing the expression in Eqn 3.3 (see Appendix II E for further details).

An alternative perspective comes from casting Eqn 3.1 in terms of an explicit spatial basis, *i.e.* $\beta = \tilde{\beta}R^T$, where the spatial basis is comprised of scaled eigenvectors of K_2 (where $K_2 = \Phi D \Phi^T = RR^T$), *i.e.*

$$R^T = D^{1/2}\Phi^T \qquad 3.4$$

which leads to the equivalent two-level model

$$\begin{aligned} Y &= X\tilde{\beta}R^T + \varepsilon_1 \\ \tilde{\beta} &= \tilde{\varepsilon}_2 \end{aligned} \qquad 3.5$$

where $\tilde{\beta} \sim N_{P,N_V}(0, S_2 \otimes I_{N_V})$ [12]. The benefit of this formulation is that different spatial priors *e.g.* based on a EGL or GGL kernels can be thought of as providing a different spatial basis set. Model comparison is then an evaluation of which basis best explains the data.

Confounds, such as scanner drift and mean signal can be accommodated conveniently into the model above by transforming the data. Consider a GLM containing two partitions; one for the signal of interest, X_1, *i.e.* experimental design matrix, and confounds, X_2

$$Y = X_1\beta_1 + X_2\beta_2 + \varepsilon_1 \qquad 3.6$$

We can use the change of variables formula (second line Eqn 3.7) to transform this into a more convenient form. Given a function of data, $r(Y)$, the log marginal likelihood is given by

[12] Note that this follows because $vec(\beta) = vec(\tilde{\beta}R^T) = (R \otimes I_P)vec(\tilde{\beta})$ and $\mathrm{cov}(vec(\beta)) = (R \otimes I_P)\,\mathrm{cov}(vec(\tilde{\beta}))(R^T \otimes I_P) = K_2 \otimes S_2$, where $\mathrm{cov}(vec(\tilde{\beta})) = I_{N_V} \otimes S_2$

$$\begin{aligned} \tilde{Y} &= r(Y) \\ p(\tilde{Y}|\alpha) &= P(Y \mid \alpha)\,|J| \\ F &= -\tfrac{1}{2}\left(\ln|\tilde{\Sigma}(\alpha)| + \tilde{y}^T\tilde{\Sigma}(\alpha)^{-1}\tilde{y} + TN\ln 2\pi - 2\ln|J|\right) \end{aligned} \qquad 3.7$$

which now includes an extra term, the Jacobian of the data transformation, $J = \left|\partial Y/\tilde{Y}\right|$. Given the transformation, $r(Y) = P_r Y P_c$ (where P_r and P_c are square matrices of size $r \times r$ and $c \times c$ respectively), its Jacobian is $J = |P_r|^{-c}|P_c|^{-r}$. If we chose $P_r = I_T - X_2(X_2^T X_2)^{-1}X_2^T$, *i.e.* the projection matrix to the null space of the confounds, and $P_c = I_{N_v}$, the model reduces conveniently to one partition

$$\begin{aligned} P_r Y P_c &= P_r X_1 \beta_1 P_c + P_r X_2 \beta_2 P_c + P_r \varepsilon_1 P_c \\ \tilde{Y} &= \tilde{X}_1\tilde{\beta}_1 + \tilde{\varepsilon}_1 \end{aligned} \qquad 3.8$$

as $P_r X_2 \beta_2 P_c = (I_T - X_2(X_2^T X_2)^{-1}X_2^T)X_2\beta_2 = 0$ and components of the second line are $\tilde{Y} = P_r Y P_c$, $\tilde{X}_1 = P_r X_1$, $\tilde{\beta}_1 = \beta_1 P_c = \beta_1$, $\tilde{\varepsilon}_1 \sim N_{r_1,c_1}(0, \tilde{S}_1 \otimes \tilde{K}_1)$, $\tilde{S}_1 = P_r S_1 P_r^T$ and $\tilde{K}_1 = P_c^T K_1 P_c = K_1$.

In this case, the Jacobian is constant and so we drop the tilde and subscript of $\tilde{X}_1$ and $\tilde{\beta}_1$ to keep the same symbols used in Eqn 3.1 (*i.e.*, by projecting the data and models onto the null space of the confounds, we can proceed as if there were no confounds). However, in general, a data transformation can be parameterized, in which case, if they are to be optimized then this term needs to be included in the objective function (Snelson et al., 2003).

3.2 The priors

In this section, we consider adaptive priors that are functions of the GLM parameters. In brief, we will assume the error or noise covariance is spatially unstructured; *i.e.*, for $\Sigma_1 = K_1 \otimes S_1$, we assume that the variance is the same over voxels, $K(\alpha)_1 = \upsilon I_{N_v}$; however,

it is easy to specify a component for each voxel, as in conventional mass-univariate analyses (we consider relaxing this assumption in the discussion). The row covariance is given by $S_1 = P_r P_r^T = P_r$, where P_r is the same as above (*i.e.* projection is an idempotent transformation). For the beta images we adopt an adaptive prior using a diffusion kernel, which is based on a WGL, as described in Chapter 2, whose column and row covariance are

$$\begin{aligned} K(\alpha)_2 &= \exp(-L\tau) \\ S(\alpha)_2 &= \eta \end{aligned} \qquad 3.9$$

In other words, the diffusion kernel, K_2, now plays the role of a spatial covariance matrix. In general, this depends on $b = \vec{\beta}$, however, we will approximate this using a constant Laplacian (see Eqn 2.35). This is pragmatic in that K_2 can be evaluated, at each step during optimization, much more simply (Harrison et al., 2007a). In our experience, WGLs based on the OLS estimate of non-smoothed data, b_{ols}, and its covariance (see Appendix II J.1) give reasonable results, as described in Chapter 4. This approximation retains the edge preserving character of the diffusive flow, without incurring the computational cost of re-evaluating the Laplacian and its eigensystem with each iteration of **EM**. However, generalizing this to update the Laplacian matrix during optimization is important (see Appendix II K) and is the focus of future work. The row covariance matrix, η, has dimensions $P \times P$, which we assume to be diagonal with non-identical components. Hyper-parameters of this model are, $\alpha = \{\upsilon, \tau, \eta\}$, where the first controls a stationary independent and identical (i.i.d.) noise component, the second the dispersion of the parameter image and third its amplitude.

The spatial covariance matrix afforded by a diffusion kernel is a very large (non-sparse) matrix covering many voxels. This means any reduction helps enormously, in terms of computational load. It can be computed efficiently using the eigenvalue decomposition of the Laplacian matrix into N_V eigenvectors and eigenvalues, as described in Chapter 2, which has the added benefit that many other computations are simplified, e.g. the determinant and trace of K_2. It also gracefully motivates a reduction by noting the eigenvalues fall off relatively quickly, due to the fact that diffusion induces smoothness

(see Figure 2-14 and note the rapid decay with larger τ). Eigenmodes with small eigenvalues contribute little to the total covariance matrix, which is the rationale for using a reduced eigensystem. This leads to the approximate diffusion kernel (c.f. Eqn 2.36)

$$K_2 \approx \sum_{i=1}^{n} g(\lambda_i)\phi_i\phi_i^T \qquad 3.10$$

where $n < N_V$. In this thesis we chose $n = N_V/10$, though, in the future we plan to optimize the number of eigenmodes using the expected value of their associated eigenvalues.

3.3 Expectation-Maximization

Inversion of the multivariate normal model in Eqn 3.2 is straightforward and can be formulated in terms of expectation maximisation (**EM**) (Dempster et al., 1977). **EM** entails the iterative application of an **E**-Step and **M**-Step, which is used to optimize the log marginal likelihood in Eqn 3.3. The **E**-Step evaluates the conditional density of the parameters in terms of their expectation and precision (*i.e.*, inverse variance); $q(b \mid y, \alpha) = N_{PN_V}(\bar{b}, \Pi^{-1})$, where

$$\textbf{E-Step} \qquad \begin{aligned} \bar{b} &= \Pi^{-1} Z^T \Sigma_1^{-1} y \\ \Pi &= \Sigma_2^{-1} + Z^T \Sigma_1^{-1} Z \end{aligned} \qquad 3.11$$

See also Appendix II D.8. The unknown covariances $\Sigma(\alpha)_i = K_i \otimes S_i$ are functions of covariance hyper-parameters, α, which are estimated by maximising the log marginal likelihood in an **M**-Step. This involves updating hyper-parameters (indexed by subscripts) by an increment, $\Delta\alpha$ (note that Δ denotes a small change here and not the Laplace operator), using a Fisher-scoring scheme[13] (see Appendix II F for more details)

[13] This is equivalent to a Newton step, but using the expected curvature as opposed to the local curvature of the objective function.

$$\Delta\alpha = I(\alpha)^{-1}\nabla_\alpha F$$

M-Step

$$\frac{\partial F}{\partial \alpha_k} = -\tfrac{1}{2} tr\left(\Sigma^{-1}\frac{\partial \Sigma}{\partial \alpha_k}\right) + \tfrac{1}{2} y^T \Sigma^{-1}\frac{\partial \Sigma}{\partial \alpha_k}\Sigma^{-1} y \qquad 3.12$$

$$I_{kl} = -\left\langle \frac{\partial^2 F}{\partial \alpha_k \partial \alpha_l} \right\rangle = \tfrac{1}{2} tr\left(\Sigma^{-1}\frac{\partial \Sigma}{\partial \alpha_k}\Sigma^{-1}\frac{\partial \Sigma}{\partial \alpha_l}\right)$$

Where $\partial F/\partial \alpha_k$ is the k^{th} element of $\nabla_\alpha F$ and I_{kl} is the (k,l) row/column of the expected Fisher Information matrix, $I(\alpha)$. In summary, to invert our model we simply specify the covariances $K(\alpha)_i$ and $S(\alpha)_i$ and their derivatives, $\partial K_i/\partial \alpha_i$ and $\partial S_i/\partial \alpha_i$. These enter an **M**-Step to provide ML-II estimates of covariance hyper-parameters. $K(\alpha)_i$ are then used in the **E**-Step to provide the conditional density of the parameters.

Pseudo-code for the algorithm is given in Figure 3-1, where prior densities are specified before optimization (top panel), *e.g.* the type of spatial prior over parameters, and the objective function (top equation) is optimized with respect to the posterior density over parameters, $q(b)$ (shorthand for $q(b \mid y, \alpha^{(m)})$), and hyper-parameters , $\alpha^{(m)}$, where the superscript indicates the m^{th} iteration (lower panel). The lower panel is comprised of **E** and **M**-Steps which are iterated until convergence. The **E**-Step updates the posterior density, with $\alpha^{(m)}$ fixed, and the **M**-Step updates α using a Fisher-scoring scheme with $q(b)$ fixed. Iterations between these two steps continues until the change in $F(q(b),\alpha)$ is below a pre-specified threshold, after which, it can be used to approximate the log-evidence of the model. This quantity is useful in model comparison and selection, as we will see later when comparing models based on different spatial priors.

We now have all the components of a generative model that, when inverted, provides parameter estimates that are adaptively smooth, with edge preserving characteristics. Furthermore, this smoothing is chosen automatically and optimises the evidence of the model. Before applying this scheme to synthetic and real data we will consider some special cases that will be compared in Chapter 4.

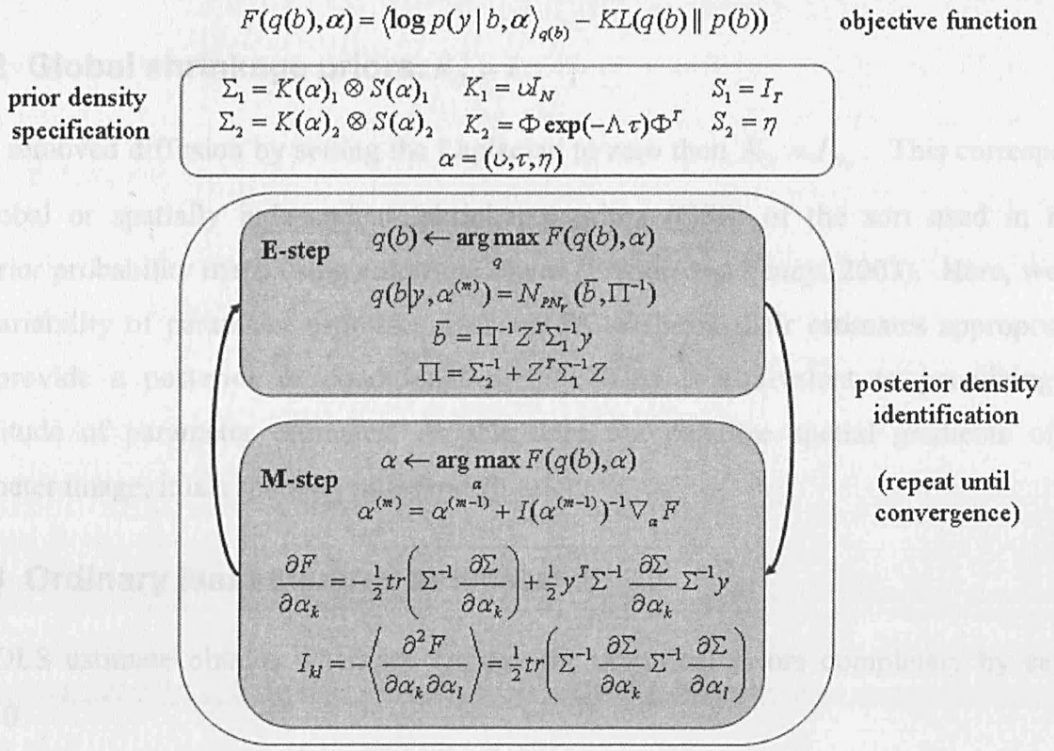


Figure 3-1: Pseudo-code of EM algorithm using Fisher-scoring scheme
The objective function to be optimized is at the top (this is equivalent to the expression in Eqn 3.3; see Appendix II E.4). Prior densities are specified before optimization (top panel) and the posterior density and point estimates of the hyper-parameters optimised by iterating **E** and **M**-steps (lower panel).

3.4 Special cases

3.4.1 Linear diffusion: $H_f = 0$

If we set the scale of the parameter dimension to zero (see Eqn 2.11), i.e. $H_f = 0$, we recover linear diffusion. The Laplacian (EGL) is now independent of GLM parameters. In this case, edges are not preserved by the smoothness prior as it is an isotropic and stationary spatial model. These kernels are useful in that they represent the isotropic and stationary assumption implicit in smoothing data with a fixed Gaussian kernel. The benefit of encoding this in an explicit spatial model is that it can be quantitatively compared to other priors (defined by different kernels), such as a non-stationary spatial model, using model evidence. This cannot be achieved if data are smoothed before entering a statistical model.

3.4.2 Global shrinkage priors: $K_2 = I$

If we removed diffusion by setting the Laplacian to zero then $K_2 = I_{N_V}$. This corresponds to global or spatially independent (shrinkage) priors (GSP) of the sort used in early posterior probability maps using empirical Bayes (Friston and Penny, 2003). Here, we use the variability of parameter estimates over voxels to shrink their estimates appropriately and provide a posterior or conditional density. This is equivalent to penalizing the magnitude of parameter estimates. As this does not penalize spatial gradients of the parameter image, it is a spatially independent prior.

3.4.3 Ordinary least squares estimate: $K_2 = 0$

The OLS estimate obtains when we remove the empirical priors completely by setting $K_2 = 0$.

3.5 Relation to other schemes

3.5.1 Restricted Maximum Likelihood

It is instructive to look at the eigenmodes of the diffusion kernel to intuit the covariance components they represent. We will do this by relating them to a restricted maximum likelihood (ReML) (Patterson and Thompson, 1974) based scheme, where the prior covariance, K_2, can be represented using n components, $\{Q_i\}_{i=1}^{n}$ (Friston et al., 2002b), i.e.

$$K_2 = \sum_{i=1}^{n} \upsilon_i Q_i \qquad 3.13$$

The weight of each component, υ_i, can then be estimated, given data, using ReML, where there are n weights or hyper-parameters to estimate. Compare this to the approximation of the diffusion kernel using n eigenmodes, where $n < N_V$, in Eqn 3.10. Here the outer product of each eigenmode can be considered as a covariance component, $Q_i = \phi_i \phi_i^T$ that is weighted by a function of the Laplacian eigenvalue, *i.e.* $\upsilon_i = g(\lambda_i, \tau) = \exp(-\lambda_i \tau)$, which is an eigenvalue of the diffusion kernel. This perspective provides a useful interpretation of

the diffusion kernel's eigenspectrum, examples of which are shown in Figure 2-14. Furthermore, it shows that our **M**-step is formally identical to ReML, when the covariance matrix is given by Eqn 3.10.

A key difference between the parameterisation of the covariance matrices in Eqns 3.10 and 3.13 is that only one hyper-parameter, τ, has to be estimated in the former. This is because a functional form (prescribed by diffusion) has been assumed over the weights. This is not the case for Eqn 3.13 where all n weights would have to be estimated separately. This could be achieved easily; however, it does not use information about the spatial process encoded in the spectrum of the Laplacian (*i.e.*, it would not conform to a diffusion prior). An additional benefit of Eqn 3.10 is that eigenmodes of a GGL represent covariance components that are informed by the (spatial) geometry of GLM parameter estimates (in our case, their OLS estimates of non-smoothed data).

3.5.2 Markov Random Fields

As seen in Eqn 2.37 the diffusion kernel is a function of the WGL eigensystem, where $g(\Lambda) = \exp(-\Lambda\tau)$. However, different functions of the eigensystem can be used to recover GMRF priors. To see this we write the prior over GLM parameters in terms of a function of the WGL eigensystem, $p(\beta) = N_{P,N_V}(0, S_2 \otimes \Phi g(\Lambda)\Phi^T)$. If we use the function $g(\Lambda) = \Lambda^{-1}$, then $\Phi g(\Lambda)\Phi^T = L^{-1}$ and we recover the spatial prior, $p(\beta) = N_{P,N_V}(0, S_2 \otimes L^{-1})$, where L is now a spatial precision matrix (Penny et al., 2005). Note that the prior in Penny *et al* is based on a spatial *precision* instead of covariance matrix, and that these authors used an isotropic Laplacian matrix. If we use the function $g(\Lambda) = \Lambda^{-2}$ we recover the bi-Laplacian precision matrix, $L^T L$. Formulating the covariance matrix of the prior over parameters as a function of the WGL eigensystem subsumes both a prior with spatial precision matrix given by the GL and diffusion-based priors given different functions of the eigenvalues.

3.5.3 Gaussian process priors

Gaussian process priors are used for nonlinear regression (MacKay, 1998; Rasmussen and Williams, 2006), where the objective is to obtain an estimate of a function, *e.g.* over space, given data at N specific locations. This leads to a simple two-level hierarchical model, similar to that in Eqn 3.2, except that the finite length vector, Zb, is replaced by a function over continuous space, $f(x)$. In other words, f has infinite dimensions, which can be modelled using a GPP, which we denote by GP to distinguish it from the finite dimensional multivariate normal density. This GPM can be expressed as

$$\begin{array}{ll} y = f + e_1 & p(y,f) = p(y \mid f)p(f) \\ f = e_2 \quad \Rightarrow & p(y|f) = GP(f, K_1) \\ e_i \sim GP(0, K_i) & p(f) = GP(0, K_2) \end{array} \qquad 3.14$$

where covariance functions are represented by K_i. The fact that a covariance *function* is used is an important feature of GPPs as it allows them to be used to make predictions at locations where a measurement has not been made. A typical covariance function of the signal is the squared exponential. In one dimension, this is

$$K_2(x,x';\alpha_2) = \upsilon_2 \exp\left(-\frac{(x-x')^2}{2\sigma^2}\right) \qquad 3.15$$

where x and x' are two points, for example, in physical space. Given N data points, Eqn 3.15 is used to compute a covariance matrix of the prior over f and typically an i.i.d noise process is assumed, *i.e.* $K_1 = \upsilon_1 I_N$. The hyper-parameters of this model are then $\alpha = \{\alpha_1, \alpha_2\} = \{\upsilon_1, \upsilon_2, \sigma\}$, which are typically optimized using the objective function

$$\begin{aligned} F(q(f),\alpha) &= -\tfrac{1}{2}\left(\ln|K(\alpha)| + y^T\Sigma(\alpha)^{-1}y + N\ln 2\pi\right) \\ K(\alpha) &= K_1 + K_2 \end{aligned} \qquad 3.16$$

which can be acheived using gradient ascent. Note that this is the same objective function used in the previous section, where $f \to Zb$ and $K_i \to \Sigma_i = K_i \otimes S_i$. The optimized model

can then be used to make predictions at test points, i.e. values of x at which data have not been observed.

Our use of spatial models, in this thesis, is different in that we do not use them to make predictions at points in the brain that have not been measured. Instead we use them to make inference about GLM parameters at measured locations, *i.e.* voxels. This means that we only require a covariance *matrix* over voxels, i.e. the matrix exponential of a WGL, instead of the more general covariance function from which a matrix can be computed. The spatial priors in this thesis are therefore finite dimensional. Note that the covariance functions used to specify GPPs could also be used in the framework proposed here. Adopting a continuous framework would be useful, particularly for interpolation, and would fit gracefully with continuous representations of diffusion on manifolds (Sochen et al., 1997). This will be the focus of future work. Next we collate the assumptions used in our implementation of the algorithm before applying it to data.

3.5.4 Summary of simplifying assumptions

The limitation of explicit spatial models of fMRI data is computational, due to the large number of voxels in a brain volume. This has led to a number of assumptions in order to increase the speed of the algorithm that can be relaxed in the future. We list the main assumptions below.

1. factorizing densities over random matrices by rows and columns and choosing a parameterized form for their covariance
2. block-diagonal approximation of WGL and reduced eigensystem to approximate the diffusion kernel for each block
3. the GGL is fixed using the OLS estimates of GLM parameters, given non-smoothed data

We will consider these in detail during the discussion. To summarize this chapter, we have described how to use the diffusion kernel of a weight graph Laplacian as the spatial covariance over voxels in a hierarchical GLM and, importantly, how its spatial scale can be optimized. Next we apply this model to synthetic and real fMRI data.

4 Application

In this chapter we use diffusion-based spatial priors to model synthetic and real fMRI data. The demonstration of each is divided into two subsections, where we present results from analyses of individual slices and volumes, which we partition into computationally manageable segments. Details about all data sets (including synthetic) are given in Appendix I. Two synthetic data sets are used that include the toy image from Chapter 2, i.e. with no temporal component, and a volume of time-series data. Three real fMRI data sets are then presented, which include two at a standard resolution (3mm^3) and one at high resolution (1mm^3). The standard resolution data are from experiments during auditory and visual motion processing. The first is single subject data, whereas the second contains twelve subjects. The high resolution data set was collected during visual stimulation. The computer and software used for these analyses were a 32-bit machine with a clock rate of 3.06GHz and 2GB of RAM and MATLAB (The MathWorks, Natick, MA) version 7.0.4.365.

4.1 Synthetic data

In this section, we use two synthetic data sets to compare the performance of three different spatial models that differed only in the form of the prior covariance over voxels; (1) a global shrinkage prior (GSP), which is spatially independent, *i.e.* $K_2 = I_{N_V}$; (2) diffusion kernel of a Euclidean graph Laplacian (EGL) and (3) diffusion kernel of a geodesic graph Laplacian (GGL). We will refer to these three models collectively as spatial models, despite the independence of the GSP. Each model was optimized using the **EM** algorithm described in Chapter 3. Comparisons include the squared error between known and estimated parameters, which we refer to as the test error (Tipping, 2004), lower bounds on the log model evidence from which Bayes factor (see Chapter 1) can be computed, OLS

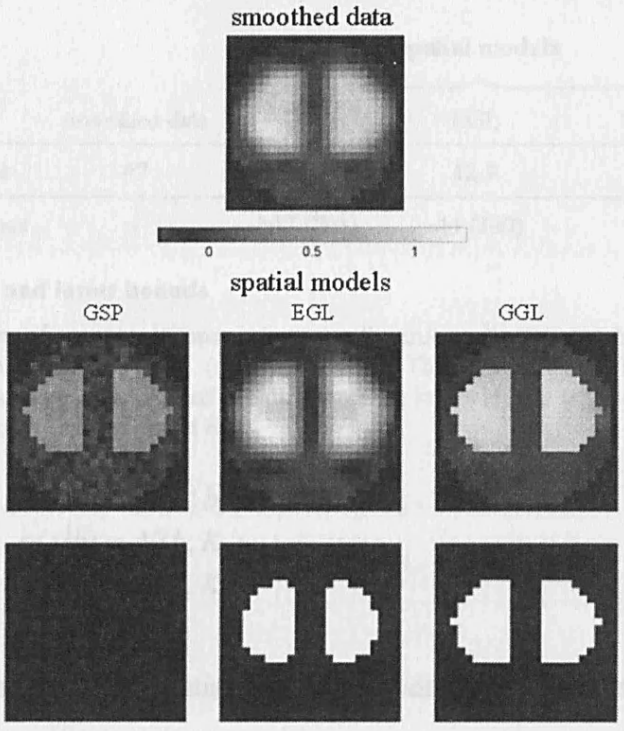


Figure 4-1: Posterior means and PPMs for synthetic image
Toy image from chapter 2 smoothed using a 2 pixel FWHM Gaussian (top; grayscale common to all parameter images) and compared to posterior mean estimates using the three spatial models below (order from left to right is GSP, EGL and GGL). Below this PPMs are shown using thresholds $p(b > 0.3) > 0.95$.

estimates given smoothed data, posterior mean parameter estimates and inferences using posterior probability maps[14] (PPMs).

4.1.1 Slice data

The synthetic 2D scalar image from Chapter 2 was used to demonstrate the denoising and edge preserving quality of the GGL-based diffusion prior and compare results with GSP and EGL-based models. The three spatial models were optimized, given the *non*-smoothed image and the objective for each model was to estimate the (known) underlying noiseless image. As such the model is very simple as the design matrix reduces to a scalar value of unity. The model is

[14] A Posterior Probability Map has two thresholds $t_1 \in \Re$ and $t_2 \in [0,1]$ that are used to show voxels where the model is at least $100 \times t_2\,\%$ certain that the effect size is greater than t_1 and is represented by the expression $p(u > t_1) > t_2$, where $u = c^T \beta$ is a contrast image, *i.e.* linear combination of GLM parameters.

		spatial models		
	smoothed data	GSP	EGL	GGL
test error	47	35.5	12.9	**0.6**
log-evidence	-	-367 (703)	-44 (380)	**335**

Table 4-1: Test errors and lower bounds

The test error is the sum of squares difference between the true and estimated image (parameter) values. The highest log-evidence was for GGL (shown in bold). The Bayes factor, i.e. ratio of probabilities between two models, is the exponential of the difference in log-evidence (shown in parentheses), which was > 100 for GGL compared to GSP and EGL.

$$\begin{array}{ll} y = b + e_1 & p(y,b) = p(y \mid b)p(b) \\ f = e_2 \quad \Rightarrow & p(y|b) = N(b, K_1) \\ e_i \sim N(0, K_i) & p(b) = N(0, K_2) \end{array}$$

where $K_1 = \upsilon I_{N_V}$ and the three spatial models only differ in terms of the form of K_2.

For comparison, the image was convolved with a 2 pixel FWHM Gaussian kernel, which is shown at the top of Figure 4-1. Note, this is not a Bayesian model as no spatial prior is used. Posterior mean estimates and PPMs, (thresholds at $p(b > 0.3) > 0.95$) are shown below. Differences are clear, with noisy estimates using GSP, blurred mean with rounded edges of high signal with EGL and preservation of edges of the underlying image using GGL. The GSP estimates demonstrate a shrinkage effect due to the prior, however, as this is spatially independent, i.e. all off-diagonals of K_2 are zero, there is no smoothing, i.e. spatial averaging between pixels. This is not the case for the diffusion-based spatial priors whose off-diagonals are in general non-zero.

The difference between EGL and GGL-based priors is best appreciated by reviewing the local kernels[15] of these priors shown in Chapter 2 (Figure 2-12 and Figure 2-13; see also contour plots of local kernels for 3D data later). The spatial profile of weights within the neighbourhood of a pixel is very different. For the EGL kernel they are the same scale for all pixels in the image. This produces estimates with the same form of smoothness as convolving the image with a fixed Gaussian kernel (see top of Figure 4-1). This is not the

[15] Where the local kernel at the i^{th} pixel is a 2-D image reconstructed from the appropriate row of K_2.

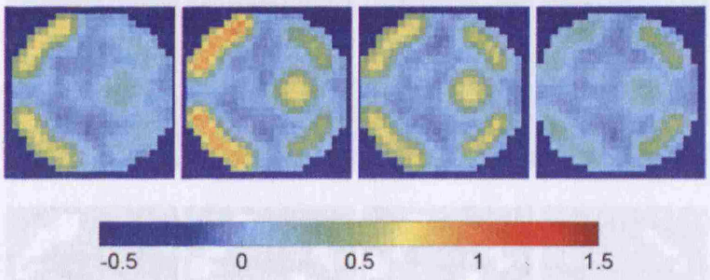


Figure 4-2: OLS estimates using smoothed data

Data were smoothed using a 2 voxel FWHM Gaussian before computing the OLS estimate for each voxel separately. Parameter estimates for the effect of interest are show for comparison with posterior mean estimates from spatial models (shown in the remaining figures of this subsection)

case for the GGL-based spatial model, where local kernels are different on either side of an edge of the image, *i.e.* the spatial model is anisotropic and non-stationary. As a result the parameter image has a variable degree of smoothness throughout, which recovers the true spatial signal without over smoothing its edges.

To quantify how well each model fitted these data we show the test error and approximate log evidence (for Bayesian models only) in Table 4-1. The test error was greatest for the smoothed data and least for the GGL-based spatial model, i.e. parameter estimates of the latter were closest (in the mean squared sense) to the true values. This is also reflected in the log-evidence, which was greatest for the GGL-based prior.

The issue with smoothing data before entering a statistical model is that there is no way to determine if the degree of smoothing is supported by the data, *i.e.* is a good assumption. By not smoothing data and using a spatial model, we can quantitatively compare different spatial models, which allow us to evaluate evidence in favour of a particular model. In the case above we can say with confidence that the data were more likely to have been generated by an anisotropic and non-stationary spatial process than one that is isotropic and stationary, which we knew to be true. We now consider a volume of time series data.

4.1.2 Volume data

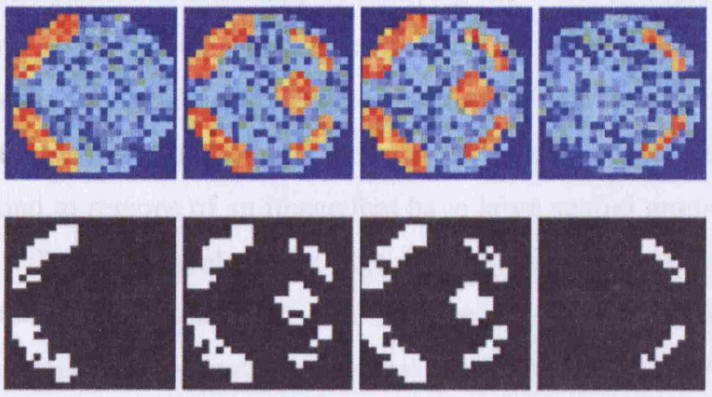


Figure 4-3: GSP-based model
Posterior means (top row) along with PPMs (thresholds $p(u > 0.5) > 0.95$)

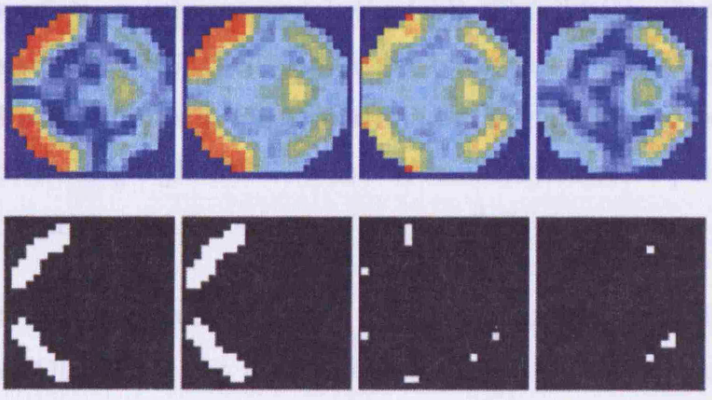


Figure 4-4: EGL-based model using full volume (no partitioning)

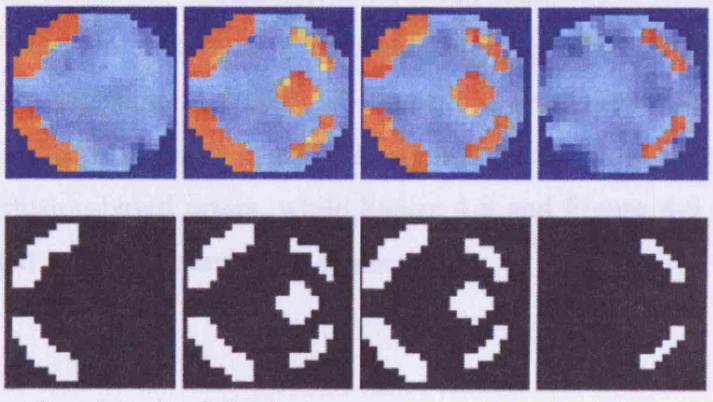


Figure 4-5: GGL-based model using full volume

Analysing a full volume is computationally demanding and is the reason for dividing a volume into segments. We do this in two ways; (1) dividing a volume into slices and (2) using the WGL to partition a volume into 3D segments. This is pragmatic, in that it can be

used to increase computational efficiency by reducing the WGL of a full volume to a block-diagonal form and analyse each block independently. The motivation for the second of these two approaches is to achieve an informed partition of a brain volume in the sense that it preserves strongly coupled nodes and predominantly removes weak edges of the graph. As these correspond to regions of an image that have large spatial gradient, for the GGL it means that a cut will tend to be along the edge dividing high/low pixel values in an image or parameters values in beta images. The purpose of this section is to compare the performance of models based on such divisions with a full volume.

The hyperparameters of each segment were optimized independently, using **EM**. This led to the comparison of seven different models (1), GSP, (2) full EGL, (3) full GGL, (4) slice-wise EGL, (5) slice-wise GGL, (6) partitioned EGL and (7) partitioned GGL. As the selection of seed points (ground nodes) determines the partition in models 6 and 7, we repeated the process using eight different sets of randomly selected points. This produced eight partitions for each model.

A volume of data was generated containing four slices (see Appendix I). For comparison, Figure 4-2 shows OLS estimates of parameters given data smoothed using a 2 voxel FWHM 3D Gaussian kernel, to represent the standard mass-univariate approach used in SPM (note that the colour bar is common to all mean estimates of these data). Figure 4-3, Figure 4-4 and Figure 4-5 contain posterior mean estimates and PPMs using the full graph (not partitioned into segments), for GSP, EGL and GGL-based spatial models respectively. Figure 4-6 and Figure 4-7 show the same for volumes divided into slices, along with local kernels of the diffusion-based priors, while Figure 4-8 and Figure 4-9 show the same for volumes segmented using EGL and GGL respectively.

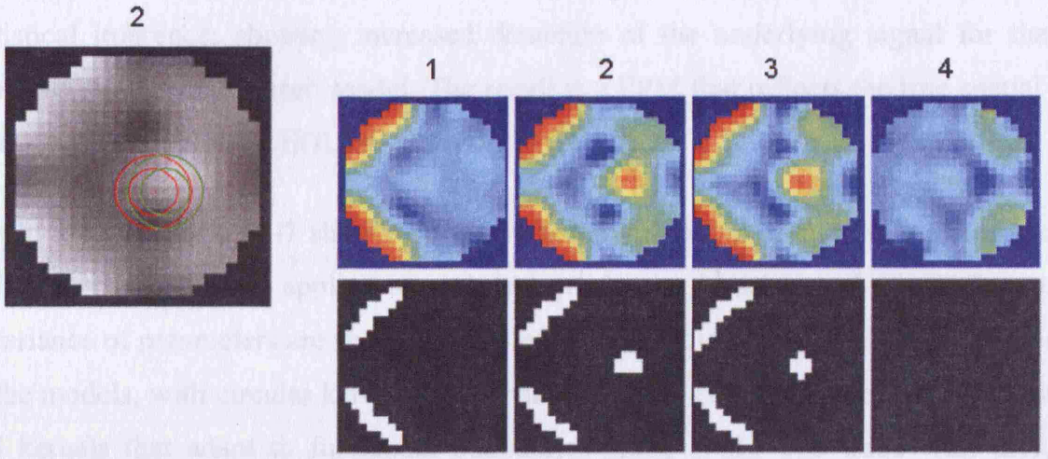


Figure 4-6: EGL-based model segmented into slices

Local kernels are shown (left) for 2 voxels

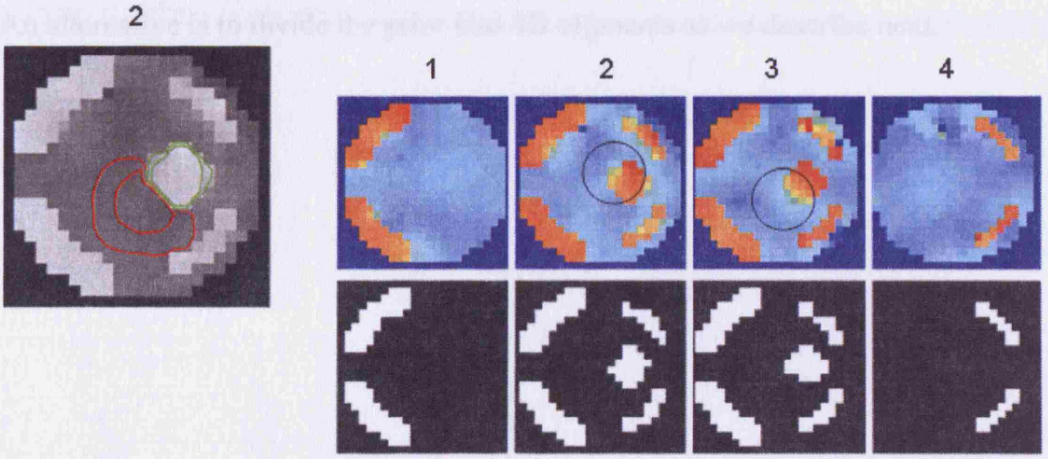


Figure 4-7: GGL-based model segmented into slices

Local kernels are shown (left) for 2 voxels. Edges of the original image that are not preserved so well are indicated by black circles in slices 2 and 3.

First we consider the full graphs in Figure 4-3, Figure 4-4 and Figure 4-5, which illustrate the performance of GSP, EGL and GGL-based spatial models for time-series data. The posterior mean estimates of the GSP prior are noisy, suggesting the benefit of using a spatial model that includes correlations between voxels. The EGL prior is isotropic and stationary, which leads to over smoothing of boundaries between high/low regions of response. This can be seen as a blurred reconstruction of the true spatial pattern of response, which occurs within and between slices. This does not occur with the GGL-based model, which preserves functional boundaries and reduces noise within homogeneous

regions. PPMs (thresholds at $p(u > 0.5) > 0.95$) are the most interesting as they represent a statistical inference; showing increased detection of the underlying signal for the GGL compared to the EGL-based model. The result is a PPM that reflects the true spatial signal more accurately than the EGL-based model.

Figure 4-6 and Figure 4-7 show results where the volume is divided into slices and EGL and GGL-based models applied to each independently. Local kernels from the posterior covariance of parameters are shown on the left. These provide insight into the [an]-isotropy of the models, with circular kernels of the same scale at all locations in the image for EGL and kernels that adapt to functional boundaries using GGL. The issue with dividing a volume into slices is that it is a somewhat arbitrary way to partition a spatial prior. As a result, edges of the original image are not all preserved (see circled regions in slices 2 and 3). An alternative is to divide the prior into 3D segments as we describe next.

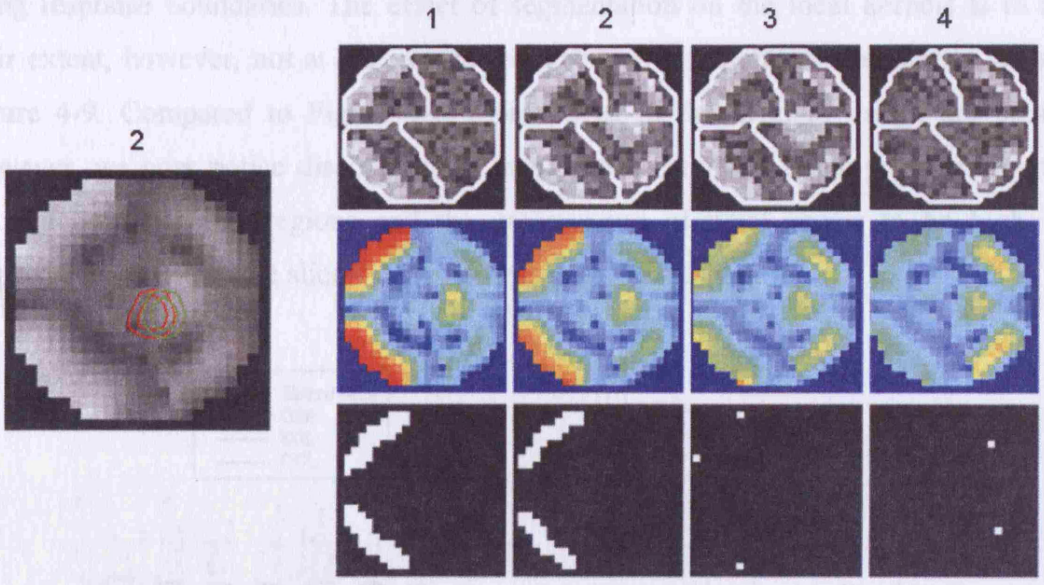


Figure 4-8: Segmenting a volume of synthetic data using an EGL-based model

Rows (from the top) show partition boundaries, posterior means and PPMs

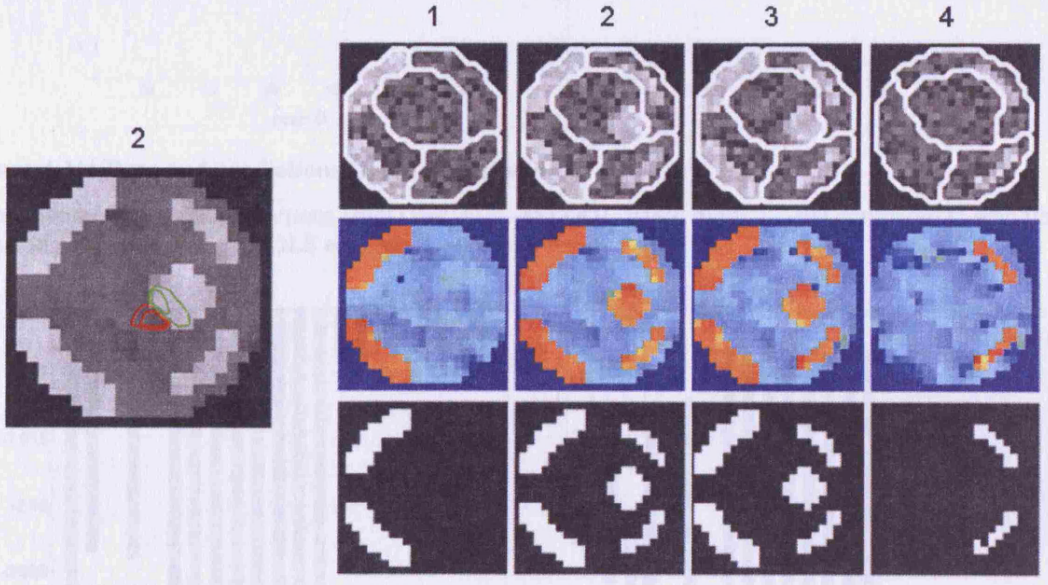


Figure 4-9: Segmenting a volume of synthetic data using an GGL-based model

Figure 4-8 and Figure 4-9 show results using the graph-Laplacian to partition the volume into 3D segments and a spatial model applied to each independently. Partition boundaries are shown (thick white lines; top), which transect a region of large response (see slices 2 and 3) for the EGL. This is not so using the GGL, which partitions the parameter image

along response boundaries. The effect of segmentation on the local kernels is to reduce their extent, however, not at the expense of adapting to functional boundaries, as seen in Figure 4-9. Compared to Figure 4-7, more edges of the original image are preserved. However, we now notice discontinuities in posterior estimates along partition boundaries through homogeneous regions and the preservation of what seems to be high spatial frequencies, or noise (see slice 4). We address in the discussion.

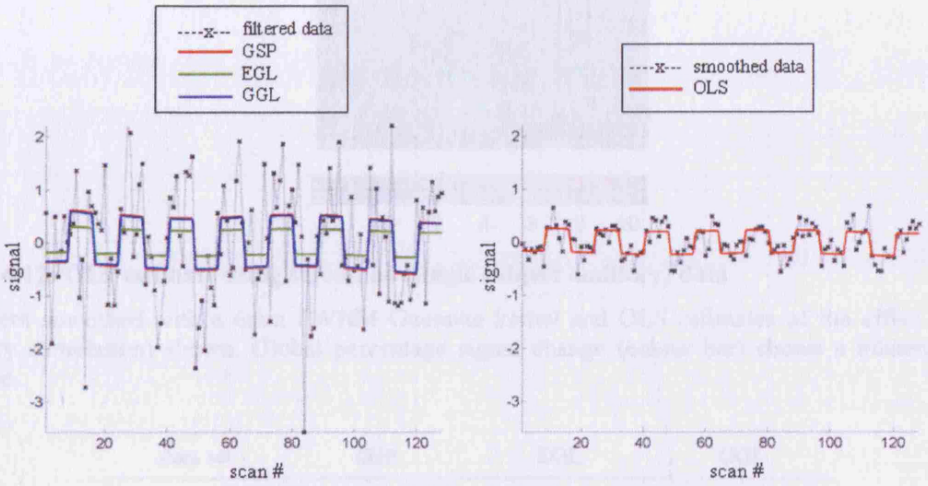


Figure 4-10: Data and predictions (synthetic volume)

Non-smoothed data and predictions from GSP, EGL and GGL-based models (left) compared to smoothed data and prediction based on OLS estimate.

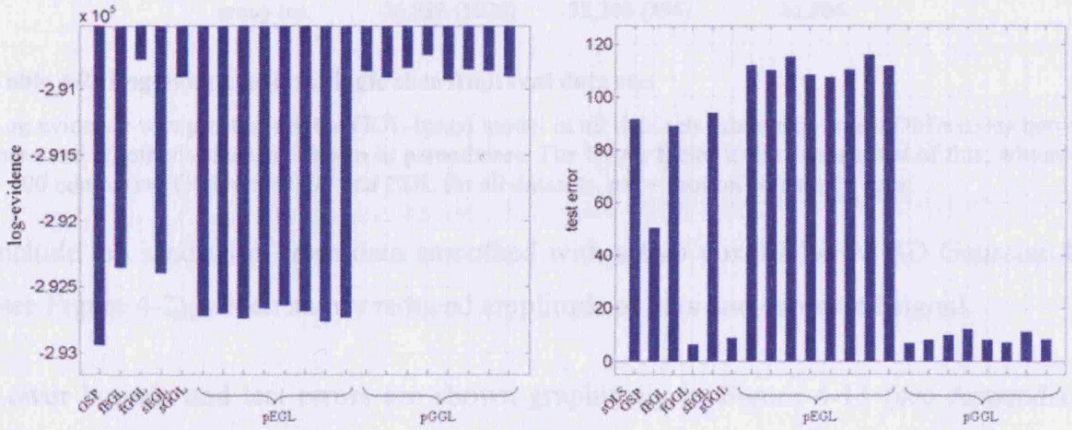


Figure 4-11: Lower bounds and test errors (synthetic volume)

The prefixes, f, s and p denote full graph, slice-wise division of a volume and partitioned volume using a GL. Note that the OLS estimate for smoothed data, denoted by sOLS, is included in the plot of test errors. The smallest difference between log-evidences (relative to the highest) was between pGGL and fGGL and was 40 (Bayes factor > 100).

Predictions (along with observed time-series) from Figure 4-8 and Figure 4-9 are shown in Figure 4-10 (left) from the marked voxel (see Appendix I). For comparison (right), we

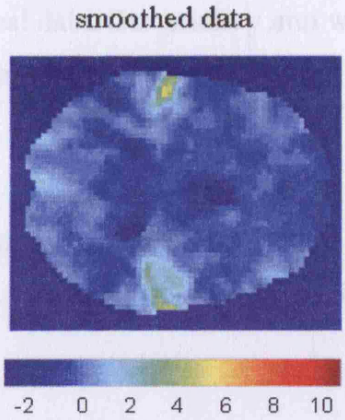


Figure 4-12: OLS estimate using smoothed (single subject auditory) data

Data were smoothed with a 6mm FWHM Gaussian kernel and OLS estimates of the effect of interest (auditory stimulation) shown. Global percentage signal change (colour bar) shows a bilateral auditory response.

data set	GSP	EGL	GGL
auditory	-28,951 (1739)	-28,474 (1262)	**-27,212**
hi-resolution	-527,375 (1515)	-529,131 (3271)	**-525,860**
single mc	90,616 (2686)	90,409 (2893)	**93,302**
group mc	-36,832 (5026)	-32,202 (396)	**-31,806**

Table 4-2: Log-evidence for a single slice from real data sets

Log-evidence was greatest for the GGL-based model in all data sets (shown in bold). Differences between these and all other values are shown in parentheses. The Bayes factor is the exponential of this, which was > 100 comparing GGL with GSP and EGL for all datasets. mc = motion coherency data.

include the prediction from data smoothed with a two voxel FWHM 3D Gaussian kernel (see Figure 4-2), which shows reduced amplitude of data and estimated signal.

Lower bounds and test errors are shown graphically in Figure 4-11 (see Appendix I for tables of all values). The greatest evidence was for a partitioned GGL-based model (pGGL). The second largest log-evidence was for the full volume (fGGL), with a difference of 40 (i.e. Bayes factor > 100). Six out of eight partitions (pGGL) had greater log-evidence than a slice-wise partition of the prior (sGGL). Interestingly, the test error was least for the

full volume, followed by pGGL. Five out of eight partitions (pGGL) had smaller test error than sGGL. These show that partitioned GGL-based models are, at least, competitive with the full volume model and at best provide a more parsimonious model.

We now compare results for real data. Our primary aim was to test whether the assumption of isotropy was supported by these data.

4.2 Real data

Here we compare models using real fMRI data from three data sets acquired during auditory and visual processing. We first consider one slice.

4.2.1 Slice data

We look first at single subject data, before turning to a group analysis.

4.2.1.1 Single subject

4.2.1.1.1 Auditory data

The auditory data set, described in Appendix I, was used to perform the same comparison as for the synthetic data of the previous section. These data were pre-processed as described in the SPM manual, with the exception of not smoothing data. A simple design matrix with two partitions (auditory stimulus and confounds) was used. This is a very simple experimental design, with the effect of interest encoded in the first column. This means that parameter estimates of this effect form a scalar field over anatomical space. For comparison, OLS estimates using data smoothed by a 6mm FWHM Gaussian kernel are shown in Figure 4-12. Posterior means of the main effect of auditory input from one slice (22 of 46) through the auditory cortex along with PPMs, thresholded to show voxels where the model is 95% certain that the effect size is greater than 2% of the global mean are shown in Figure 4-13 for GSP, EGL and GGL-based spatial priors respectively.

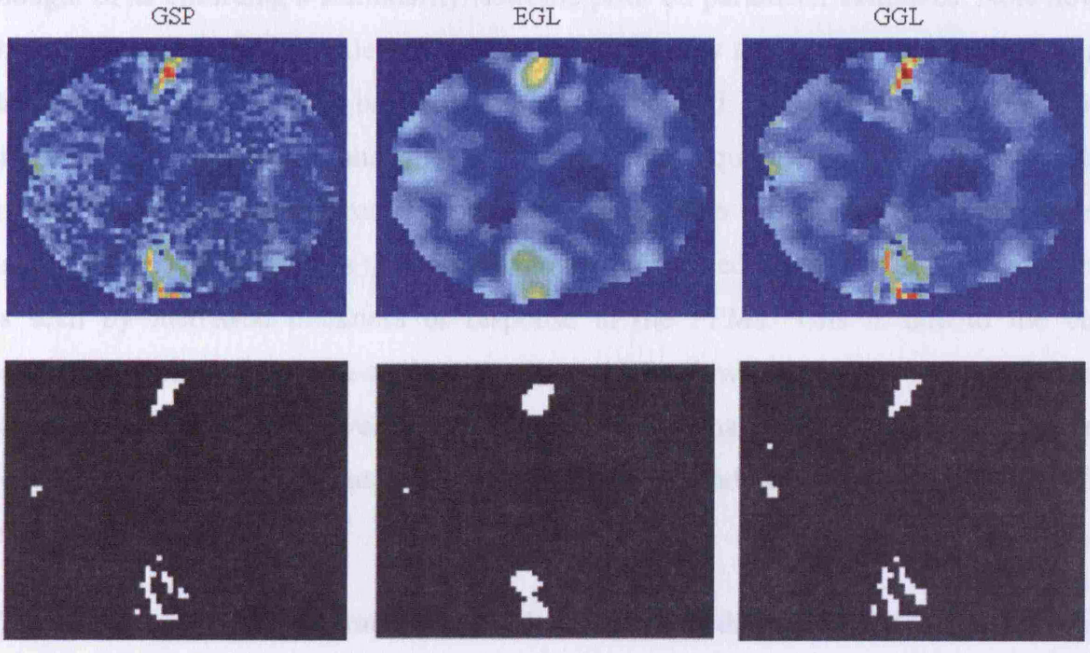


Figure 4-13: Posterior means (top) and PPMs for auditory data

Statistical images from the three spatial models (GSP, EGL and GGL from left to right; see colour scale in Figure 4-12). Bilateral activations are detected in all three, however, the fine detail of response is lost using the stationary and isotropic model (EGL). Thresholds for PPMs are $p(u > 2) > 0.95$.

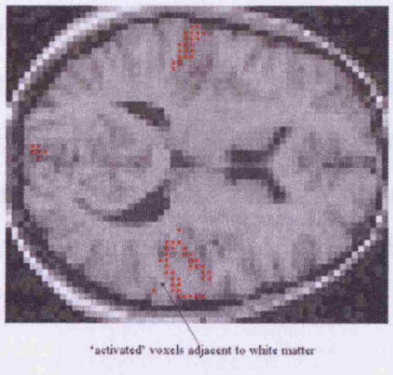


Figure 4-14: PPM from GGL-based spatial model overlaid on structural MRI of subject

The anatomical image is at the same resolution as functional data. Active voxels (red) lie along grey matter adjacent to white matter (lighter regions of anatomical image)

As for the synthetic data, differences in estimated responses are clear. Those for the GSP model are noisy, however, have structure, with activation in bilateral cortices, which is also seen in the PPM below. The EGL-based model produces smooth parameter images, where activations have been reduced to 'blobs'. This is similar to OLS estimates (Figure 4-12) where data has been smoothed, i.e. smoothing data with a fixed Gaussian kernel can be

thought of as enforcing a stationarity/isotropic prior on parameter estimates. Note however that in this approach the scale of the kernel is *chosen* by the user and not estimated given data. In contrast to the EGL-based spatial model, the GGL model shows less attenuation of signal at peaks of response and smooth estimates within quiescent regions. This is reflected in the PPM, which shows preservation of structure within activated regions, similar to that using GSP. The difference is that more structure is detected using the GGL diffusion kernel as seen by increased thickness of response in the PPMs. This is due to the border-preserving nature of the non-stationary prior, which allows the degree of smoothness of a parameter image to vary over space. The result is a sharper parameter image, as edges between functionally segregated regions are preserved and not blurred by the constraint of isotropy and stationarity.

The difference in PPMs is crucial as decisions regarding data are based on such inferences. The PPM using a GGL is shown in Figure 4-14 overlaid on an anatomical image (at the same resolution as functional data). White matter has, in general, a lighter shade in this image. The figure shows 'activations' adjacent to white matter and concurs qualitatively with our expectation that BOLD signal has a cortical origin[16]. Log-evidences are shown in Table 4-2, which supports the non-stationary model over the other two (Bayes factor > 100). This model was able to extract the structured deployment of cortical responses that are otherwise blurred by EGL. Note that this comparison could not have been made if data were smoothed before entering a statistical model. Contour plots of local kernels are shown for EGL and GGL-based models in Figure 4-15, which illustrates the istropic/stationary and anisotropic/non-stationary nature of these models.

[16] Note, however, that the neural basis of the BOLD response is still an active area of research (Logothetis and Wandell, 2004; Nair, 2005).

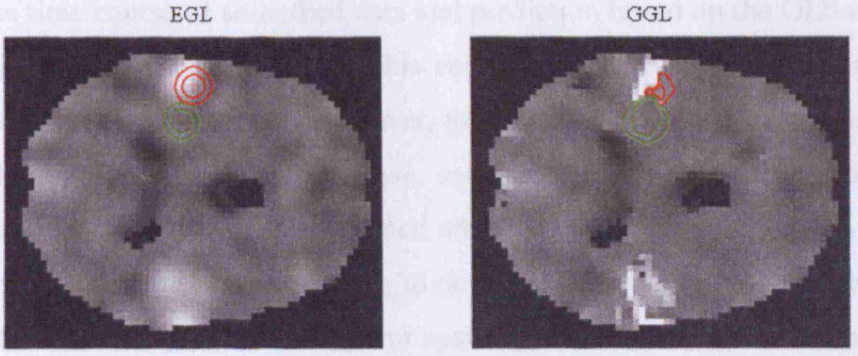


Figure 4-15: Local kernels of EGL (left) and GGL-based spatial models (auditory data)

A row of the diffusion kernel reformatted as an image (referred to as a local kernel) shows the spatial distribution of weights of the model for two voxels (best seen at the center of contours on the left). This is overlaid on the posterior mean estimate for each spatial model.

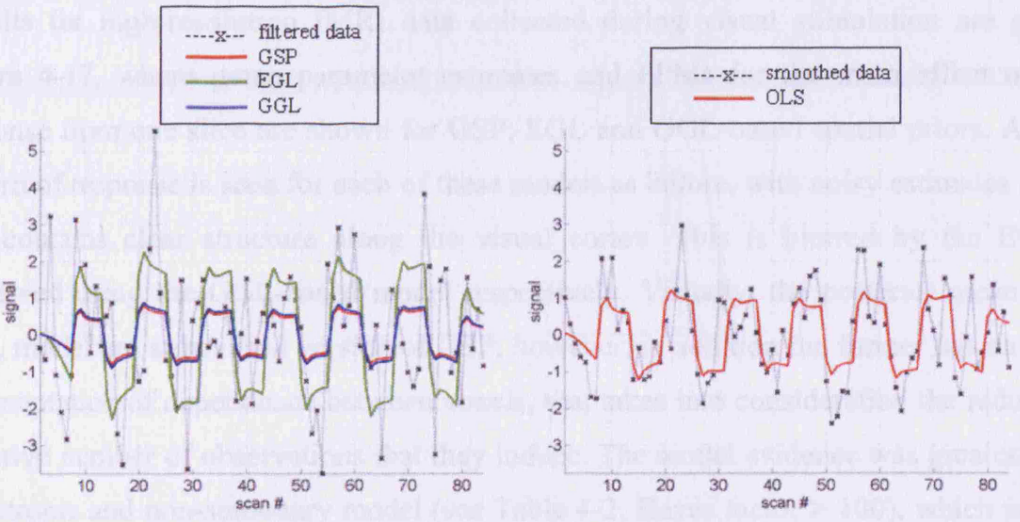


Figure 4-16: Data and predictions from one voxel (auditory data)

Traces are from the marked voxel in Figure 4-15 (red). (left) non-smoothed data and predictions from three spatial models (GSP, EGL and GGL) and (right) smoothed data and prediction given OLS estimate.

Predictions from GSP, EGL and GGL-based models are shown in the left panel of Figure 4-16 from a voxel at the boundary of response in the left auditory cortex (at the centre of the red local kernel in Figure 4-15). These show a poor fit for EGL, in that the predicted response is much greater than the data can support, suggesting that the assumption of isotropy is inappropriate for these data. We understand why this is so by considering the local kernels (left) of Figure 4-15, i.e. the spatial region of weights that determine smoothness is the same for all voxels. That is, they do not depend on values of the beta field, which leads to averaging over space that blurs functional boundaries. We have

included the time course of smoothed data and prediction based on the OLS estimate on the right of Figure 4-16, for comparison. This shows how data have been regularized by the pre-processing step of smoothing. However, this comes at the expense of blurring the data, which leads to a larger predicted response, compared to using non-smoothed data and the GGL-based model, at this voxel. Note, that after data have been smoothed it is difficult to determine whether this was a good thing to do. We can only make such statements if non-smoothed data are analysed with different spatial priors that encode assumptions as to the nature of the generative process responsible for the data.

4.2.1.1.2 High resolution

Results for high-resolution fMRI data collected during visual stimulation are given in Figure 4-17, where mean parameter estimates and PPMs for the main effect of visual response from one slice are shown for GSP, EGL and GGL-based spatial priors. A similar pattern of response is seen for each of these models as before, with noisy estimates for GSP that contains clear structure along the visual cortex. This is blurred by the EGL and preserved using the GGL-based model respectively. Visually, the posterior means of the GGL model are a denoised version of GSP, however, in addition the former has an explicit representation of dependence between voxels, that takes into consideration the reduction in effective number of observations that they induce. The model evidence was greatest for the anisotropic and non-stationary model (see Table 4-2; Bayes factor > 100), which is able to extract a shape of cortical response that fits with known neuroanatomy.

Predictions from the marked voxel (cross in top row centre of Figure 4-17), within a region of large response, are shown in the left panel of Figure 4-18 for all three models along with non-smoothed data from that voxel. This shows similar predicted responses for GSP and GGL-based models, which (visually) provide a good explanation of the data. In contrast, the prediction using the EGL-based model is poor. A similar effect is seen in the right panel for data that has been smoothed with a 2 voxel FWHM fixed Gaussian kernel, which again shows how this pre-processing step results in data that is regularized, but at the expense of its magnitude.

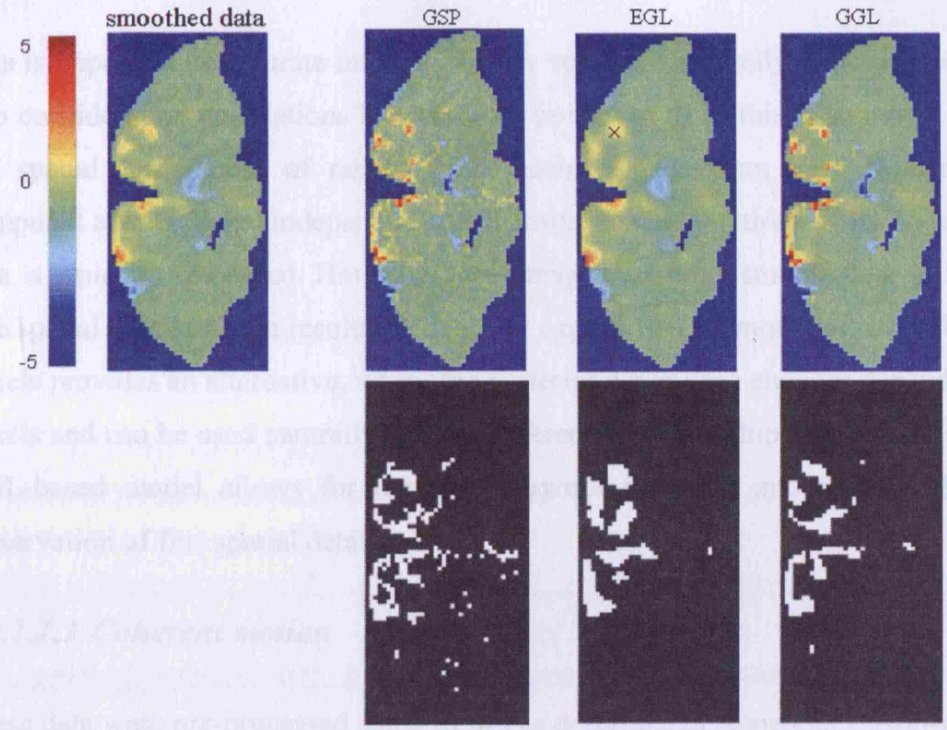


Figure 4-17: Posterior means (top) and PPMs for (high-resolution) data

OLS parameter image (top left) using data smoothed using a 2mm FWHM Gaussian is shown (percent signal change indicated by the colour bar) for comparison with posterior mean images of the main effect (visual stimulation) from the three spatial models (top row) and PPMs ($p(u > 2) > 0.95$).

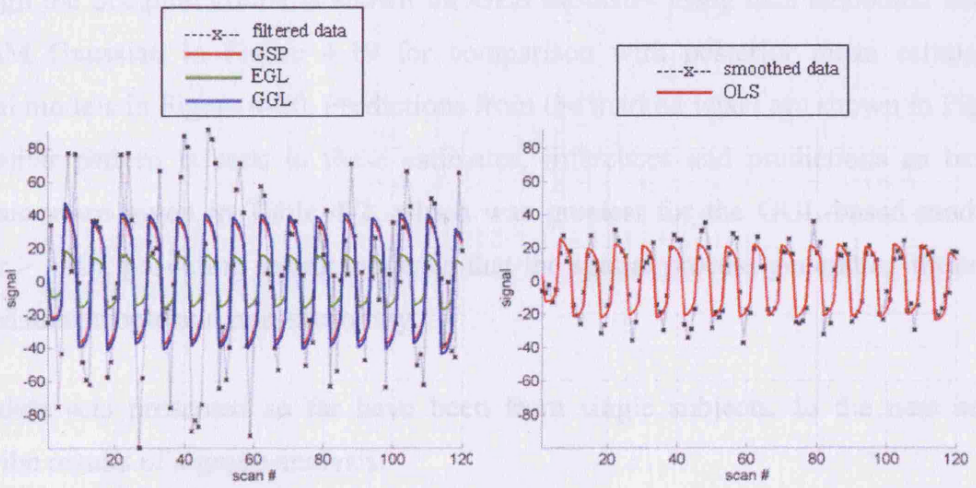


Figure 4-18: Data and predictions from the marked voxel in Figure 4-17

Non-smoothed data (left) along with predictions from the three spatial models compared to smoothed data and prediction given OLS estimate (right).

This is important as accurate inference over a volume, i.e. family of voxels, requires taking into consideration correlations between data points. In SPM this is achieved by estimating the spatial smoothness of residuals and using results from RFT to correct p-values computed at each voxel independently. To ensure the assumptions of RFT are not violated, data is typically smoothed. However, neuroimagers may not smooth data so as to preserve fine spatial detail in high resolution data. An explicit spatial model of dependence between voxels provides an alternative, where the posterior covariance encodes dependence between voxels and can be used naturally to make inferences over multiple voxels. In particular, the GGL-based model allows for non-stationary/non-isotropic smoothness, which leads to preservation of fine spatial detail.

4.2.1.1.3 Coherent motion

These data were pre-processed using SPM5 as described in Appendix I. Non-smoothed data were analysed for the three spatial models as above. The difference in this model was that the design matrix contained two columns representing onsets of motion and stationary stimuli. This means that a vector-field of GLM parameters now has to be estimated. A contrast image, comparing the effect of motion to stationary stimuli along with a PPM through the occipital cortex is shown for OLS estimates using data smoothed with a 6mm FWHM Gaussian in Figure 4-19 for comparison with posterior mean estimates using spatial models in Figure 4-20. Predictions from the marked voxel are shown in Figure 4-21. A similar pattern is seen in these estimates, inferences and predictions as before. Log evidences are given in Table 4-2, which was greatest for the GGL-based model (Bayes factor > 100), providing strong evidence that the spatial process generating these data was indeed anisotropic and non-stationary.

The data sets presented so far have been from single subjects. In the next section we describe results of a group analysis.

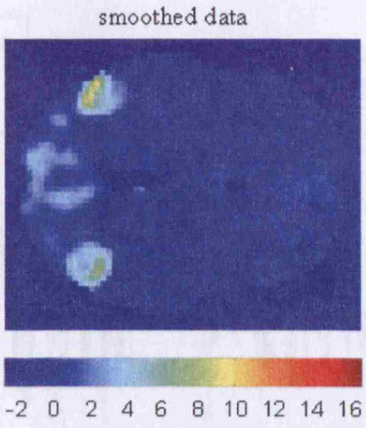


Figure 4-19: OLS estimates using smoothed (single subject mc) data

Contrast of responses to moving and stationary stimuli during a motion coherency (mc) study.

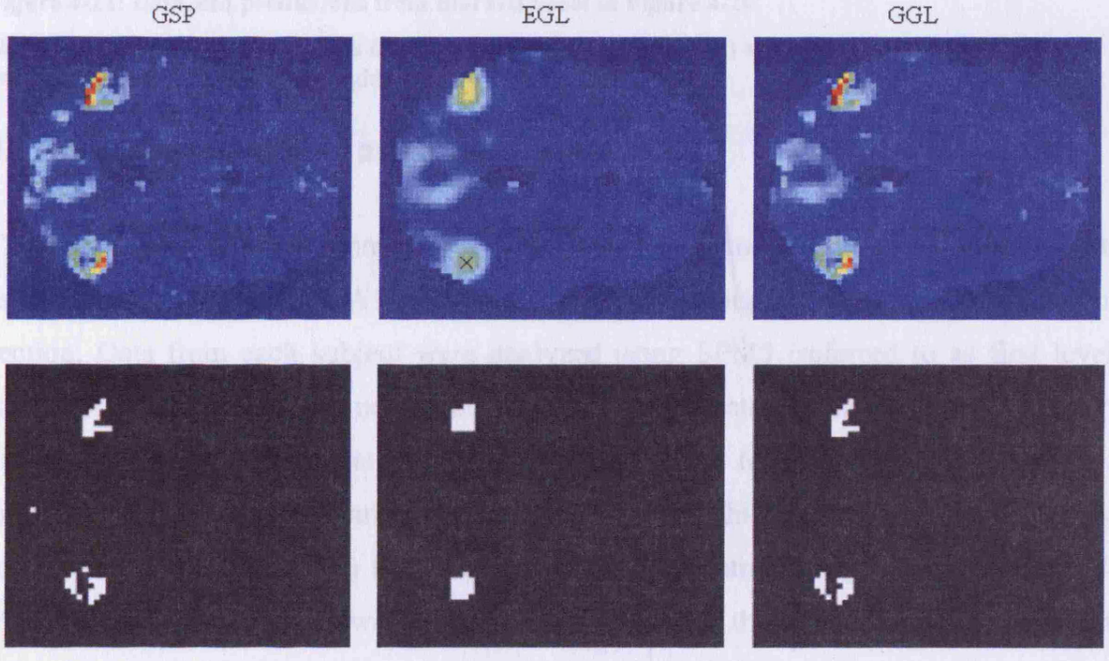


Figure 4-20: Posterior means (top) and PPMs for (single subject mc) data

PPM thresholds $p(u > 2) > 0.95$

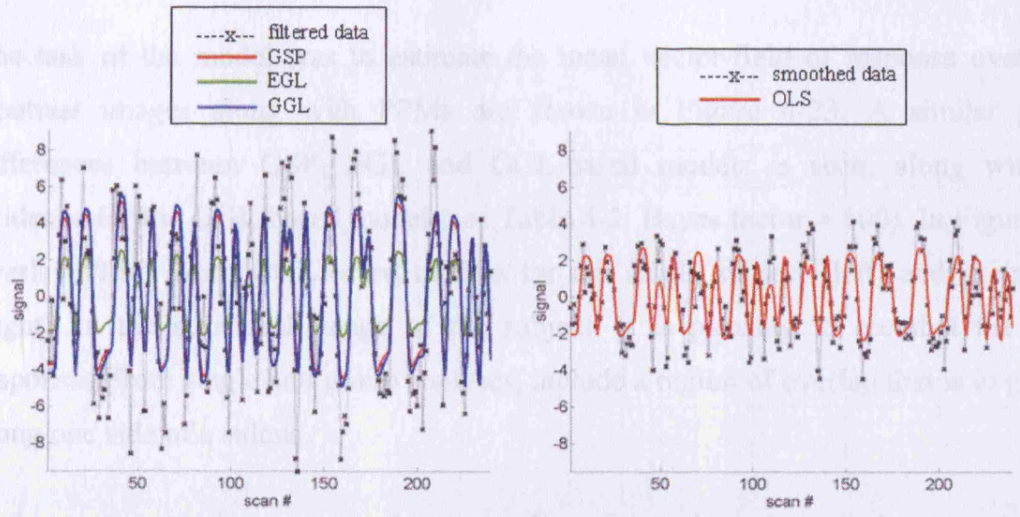


Figure 4-21: Data and predictions from marked voxel in Figure 4-20

Non-smoothed data and predictions from the three spatial models (left) and smoothed data along with prediction using OLS estimates (right)

4.2.1.2 Random-effects analysis

These data were collected from twelve subjects during a study of the visual motion system, as described in Appendix I. A single subject from this group was presented in the previous section. Data from each subject were analyzed using SPM5 (referred to as first level or within subject analysis) on non-smoothed data to generate contrast images of the main effect of moving verses stationary stimuli. We used two temporal basis functions in this analysis; the canonical HRF and its temporal derivative. This produced two contrast images (one for each basis function) for each subject. These contrast images comprised the data, Y, for a second level (or between subject) analysis and is the standard approach to random effect analysis in SPM (see Chapter 12 in (Friston et al., 2006)). This meant that there was a vector-field of data, comprised of contrast images from a first level analysis, for each subject. For comparison, data from each subject was smoothed with a 6mm FWHM Gaussian kernel and contrast images smoothed again with the same kernel, as is standard practice. A contrast image (sum of response from each basis) of OLS estimates, through the same slice as for the single subject analysis (Figure 4-19) is shown in Figure 4-22.

The task of the model was to estimate the mean vector-field of response over subjects. Contrast images along with PPMs are shown in Figure 4-23. A similar pattern of differences between GSP, EGL and GGL-based models is seen, along with highest evidence for the GGL-based model (see Table 4-2; Bayes factor > 100). In Figure 4-24 we overlay PPMs from GGL-based models for the single subject (left) and group analysis (right) on the structural image of the subject. It is pleasing to see that the estimated responses, from single and group analyses, include a region of overlap that is in grey matter along one side of a sulcus.

In this section, we have analyzed just one slice of data. Including all slices in an analysis is a challenge due to the large number of voxels in a brain volume. We achieve this by dividing the volume into computationally manageable segments, which we consider next.

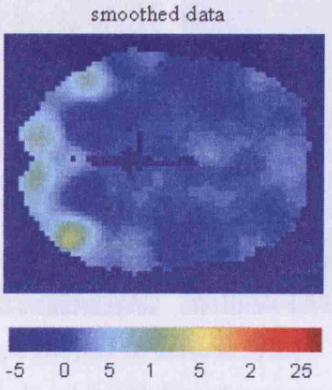


Figure 4-22: OLS estimate using smoothed data (group mc study)

Contrast image of OLS estimates from a second level (between subjects) analysis (12 subjects). Data from each subject were smoothed with a 6mm Gaussian. Contrast images were smoothed further using the same scale kernel before OLS estimation.

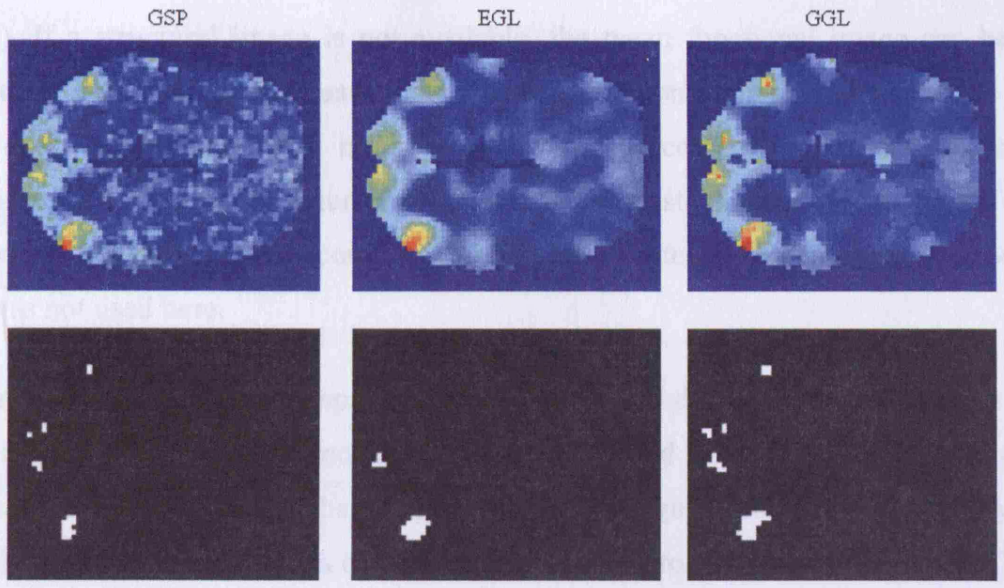


Figure 4-23: Posterior means (top row) and PPMs for group (mc data) analysis

Thresholds for PPM set at $p(u > 8) > 0.95$. mc = motion coherency.

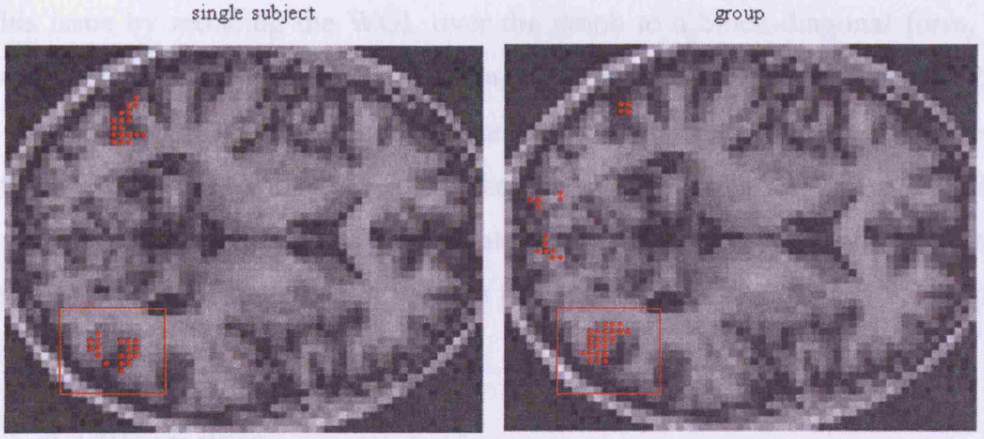


Figure 4-24: PPMs overlaid on single subject structural MRI

PPMs from GGL-based models of single subject (left) and group motion coherency data (see Figure 4-20 and Figure 4-23) are compared.

4.2.2 Volume data

In this section, we use a WGL to partition a brain volume as for the synthetic data. This volume can be defined in a number of ways. We used two approaches, based on; (1) tissue probability maps computed using a structural image of a single subject (see Figure 2 Appendix I) and (2) the mean functional image. The first allows for an anatomically informed selection of voxels, in particular, excluding those from cerebral spinal fluid

(CSF). If a structural image is not available, the mean functional image can be used to provide a mask, but has the disadvantage of including non-brain structures such as the scalp and eyes. However, these masks can be pre-processed to provide a reasonable representation of a brain volume or volume of interest. Alternatively a mask could be defined by thresholding an F-contrast of the effects of interest from a standard SPM, though this was not used here.

This mask in turn defines the spatial extent of functional data to be included in the analysis. In particular, it specifies the node set of the graph used to construct the spatial prior, i.e. diffusion kernel. These graphs, in general, have irregular boundaries and may contain "holes", e.g. excluding regions containing CSF, similar to the example shown in Chapter 2. The advantage of using such a graph is that the size of the spatial covariance matrix is reduced, which is then easier to compute. The disadvantage is that irregular graphs are more computationally demanding to compute with compared to regular graphs. We deal with this issue by reducing the WGL over the graph to a block-diagonal form, which we achieve using the isoperimetric partitioning algorithm described in Chapter 2 to select nodes to be included within a block (i.e. segment). This selection can be informed of the strength of edges between nodes, such that partition boundaries will predominantly be along weak connections. This has the advantage of keeping strongly coupled nodes within a segment, as we demonstrated for a volume of synthetic time-series earlier, and present for real data next.

4.2.2.1 Auditory data

A region of interest (ROI) was selected that included activations from both auditory cortices and a similar analysis as for the synthetic volume performed. Log-evidence plots are shown in Figure 4-25 (right). These show greater evidence for fGGL (see figure caption for abbreviations) over fEGL and sGGL over sEGL. Partition boundaries through one slice overlaid on the OLS estimates for EGL and GGL-based partitions are shown in Figure 4-25 (left). Note that this is a cross-section, where each segment can, in general, be an arbitrary shape as long as it is connected. The difference between these partitions is easily seen. Log-evidence for all models is shown on the right which shows, on average, greatest

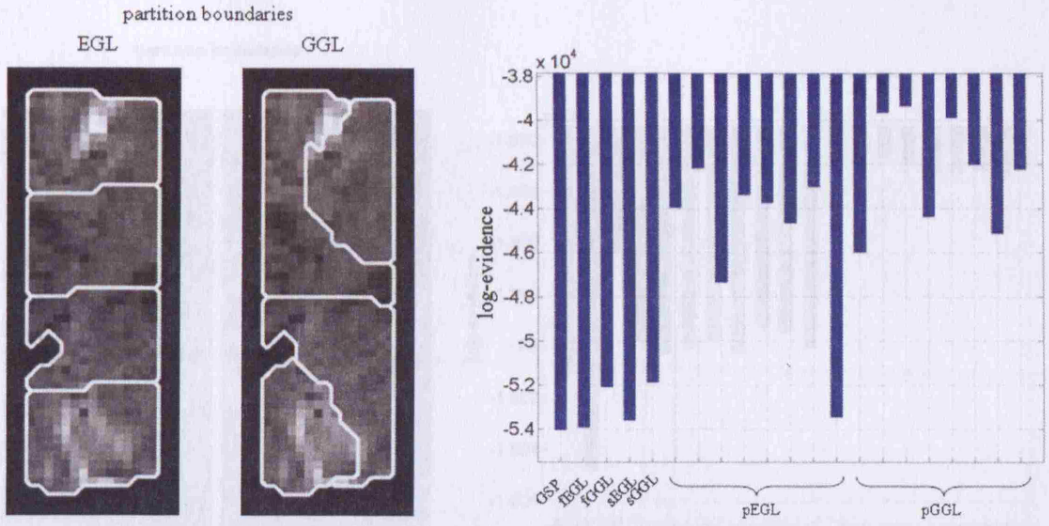


Figure 4-25: Partitioned ROI (auditory data) and lower bounds

Slice through a ROI (left) that has been partitioned using a EGL and GGL. Log-evidence plots for all models (right). The prefixes, f, s and p denote full graph, slice-wise division of a volume and partitioned volume using a GL. The smallest difference in log-evidence was between pGGL and pEGL and was $\sim 3\times10^3$ (Bayes factor > 100).

evidence for the pGGL model, though this is not so for all partitions. The second largest log-evidence was for the pEGL model, with Bayes factor > 100. Of interest is the poor performance of the full GGL model compared to pGGL, which is due to variability in the observation error in the volume.

4.2.2.2 High resolution

A mask was computed from the mean functional image. This contained many non-brain voxels due to attenuation of signal with increasing depth into the brain. These regions were eroded from the original mask to produce a more suitable volume that contained only brain tissue. A cross section through partition boundaries generated using EGL and GGL is shown in Figure 4-26 (left). Differences are clear with partition boundaries tending to be along edges of the parameter image for the GGL-based partition (see circled region). A plot of log evidences for all models is shown on the right, which shows improved fit of the partitioned GGL model compared to all others (Bayes factor > 100).

4.2.2.3 Motion coherency data

A restricted volume comprised of four slices and a full volume of the single subject and group were analyzed as above. Posterior mean estimates through the same slice used in the

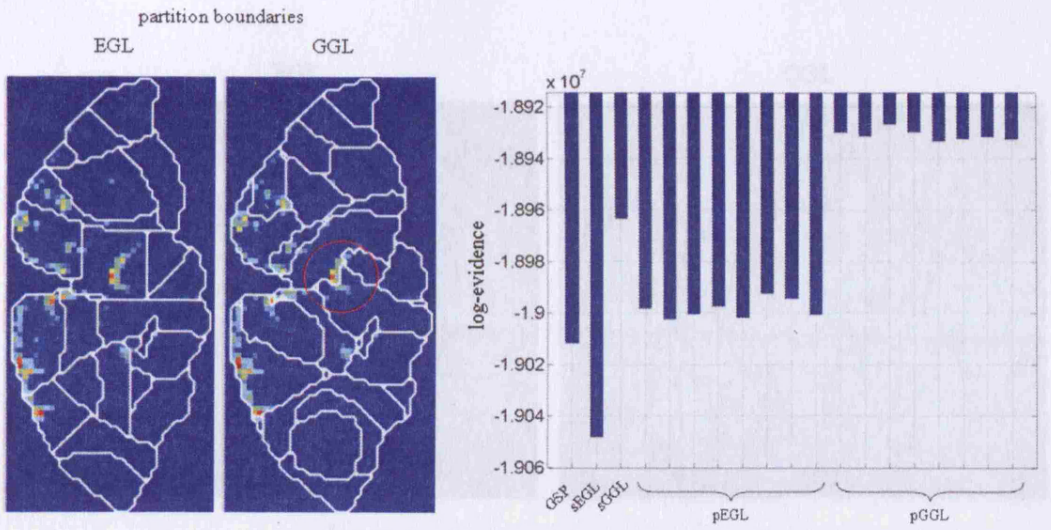


Figure 4-26: Partition boundaries for high-resolution data and lower bounds

Same layout as previous figure. The smallest difference in log-evidence was between pGGL and sGGL and was $\sim 3\times10^4$ (Bayes factor > 100).

first section of this chapter and log-evidence plots are shown for the two volumes in Figure 4-27, Figure 4-28, Figure 4-29 and Figure 4-30 for single subject and group analysis respectively. We notice the familiar differences between EGL and GGL-based models, *i.e.* blurred responses on the left compared to right. On average, the partitioned GGL-based model is supported most by all data sets, though this is not so for all partitions of the single subject (full volume) and group analysis (restricted volume) (see Figure 4-28 and Figure 4-29). This is interesting and speaks to the importance of relaxing the current assumptions such as using a fixed GGL. Also we note the impression of partition boundaries (similar to synthetic data), in particular, for pEGL Figure 4-28 (left), which suggests using soft instead of hard partition boundaries, i.e. overlapping segments. We will address these issues in the discussion.

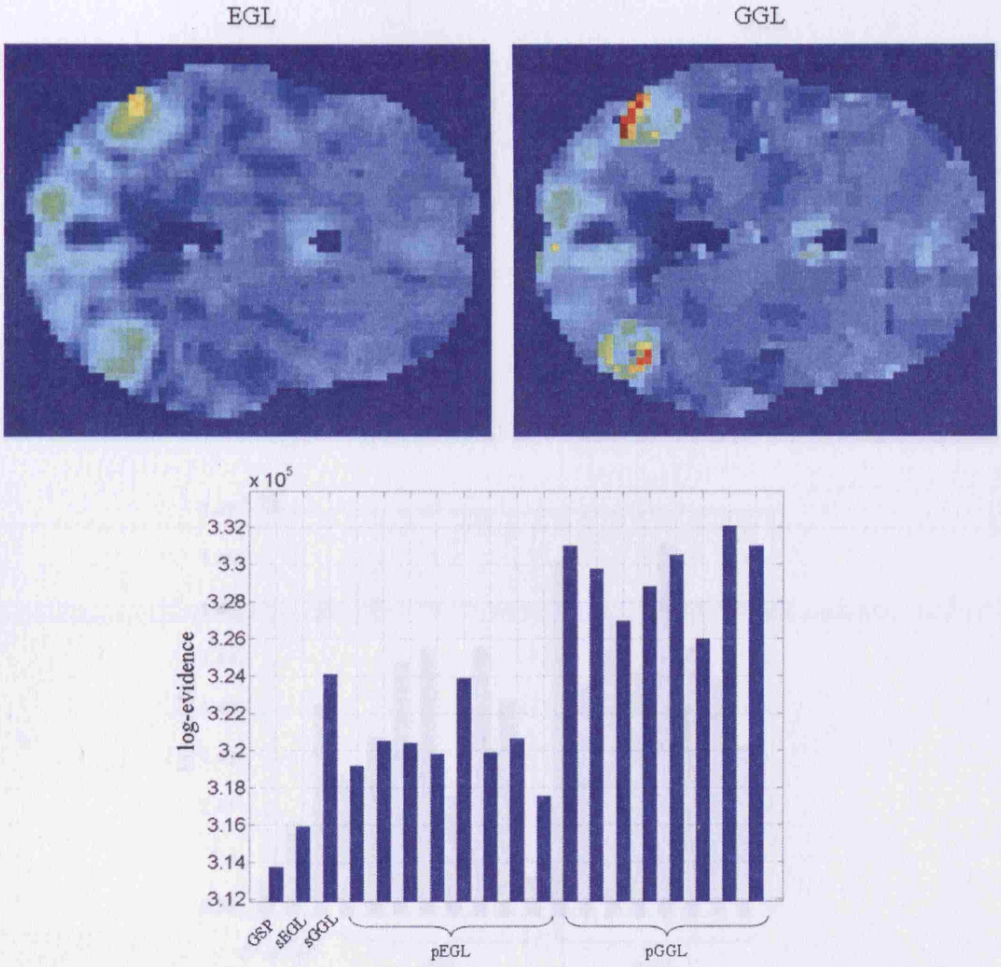


Figure 4-27: Posterior means and lower bounds (restricted volume; single subject mc)

The smallest difference in log-evidence (relative to the largest value) was between pGGL and sGGL and was ~8×10^3 (Bayes factor > 100). mc = motion coherency data.

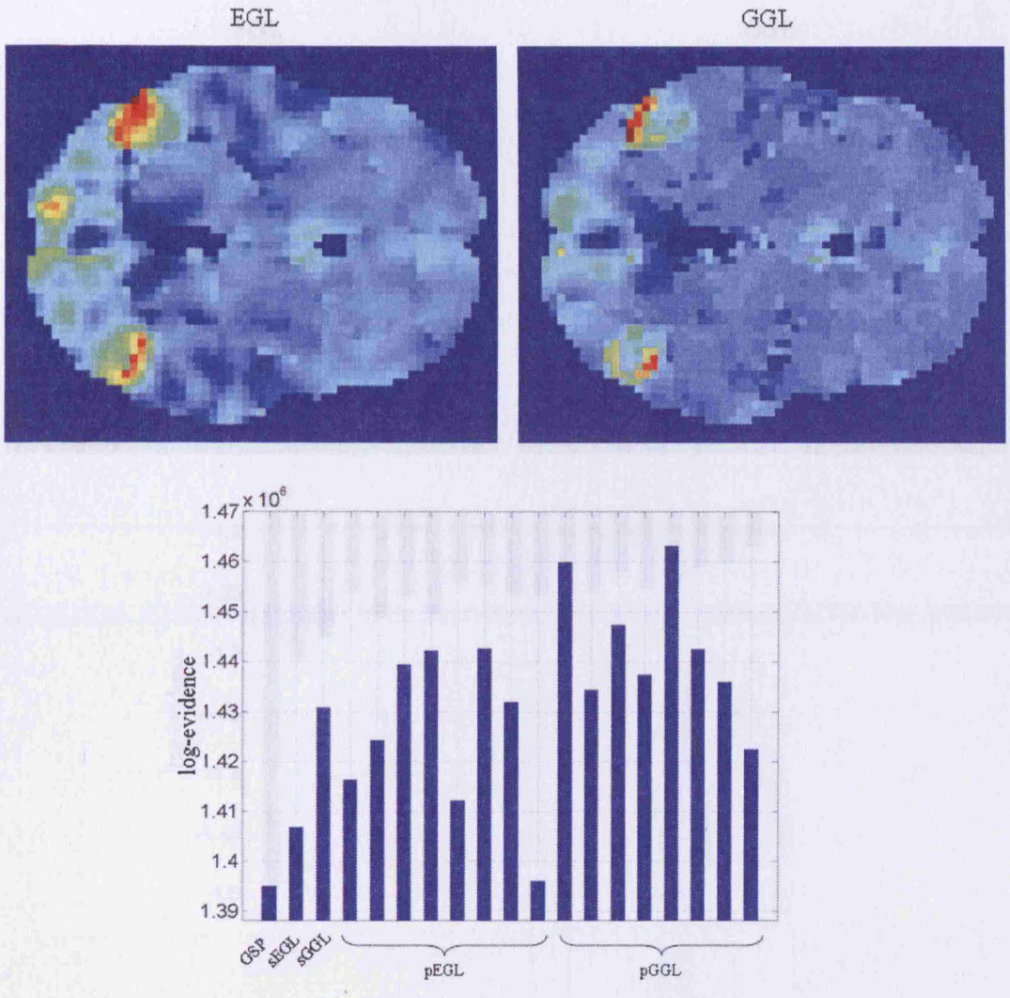


Figure 4-28: Posterior means and lower bounds (full volume; single subject mc)

The smallest difference in log-evidence (relative to the largest value) was between pEGL and pGGL and was ~2×10^4 (Bayes factor > 100).

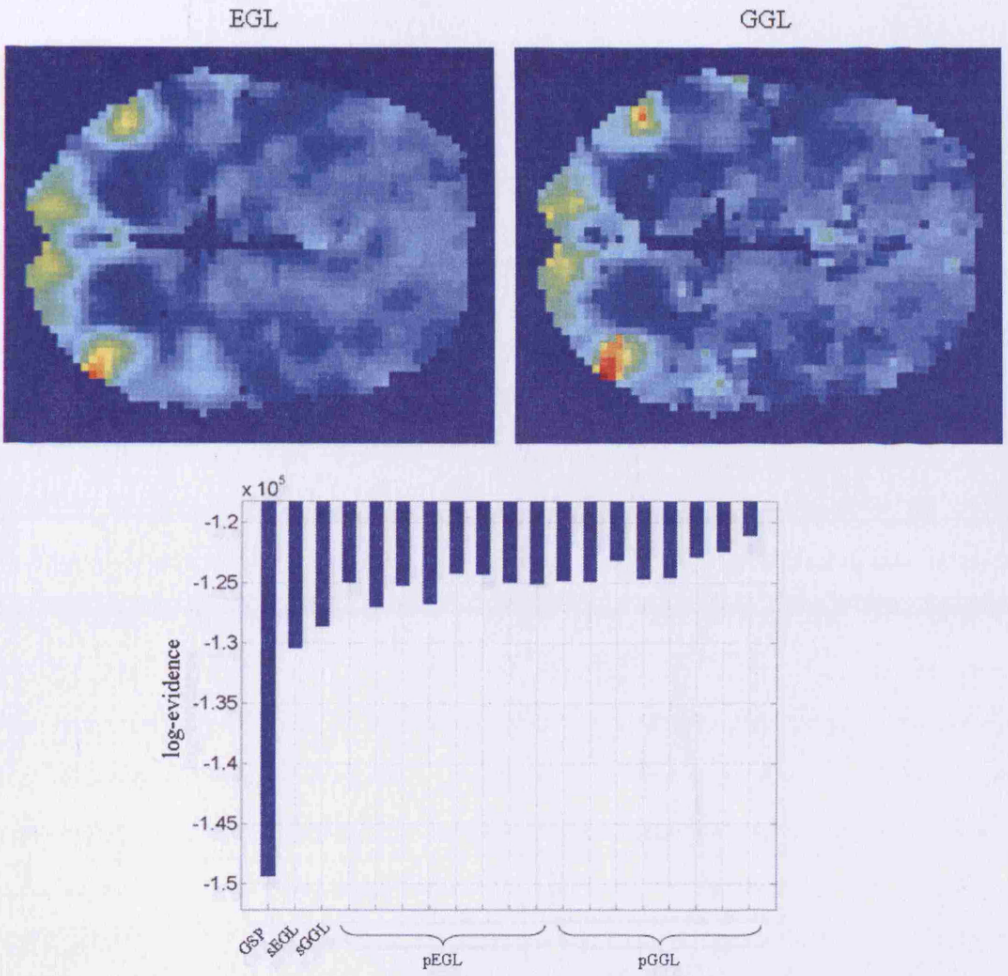


Figure 4-29: Posterior means and lower bounds (restricted volume; group mc)

The smallest difference in log-evidence (relative to the largest value) was between pGGL and pEGL and was $\sim 3 \times 10^3$ (Bayes factor > 100).

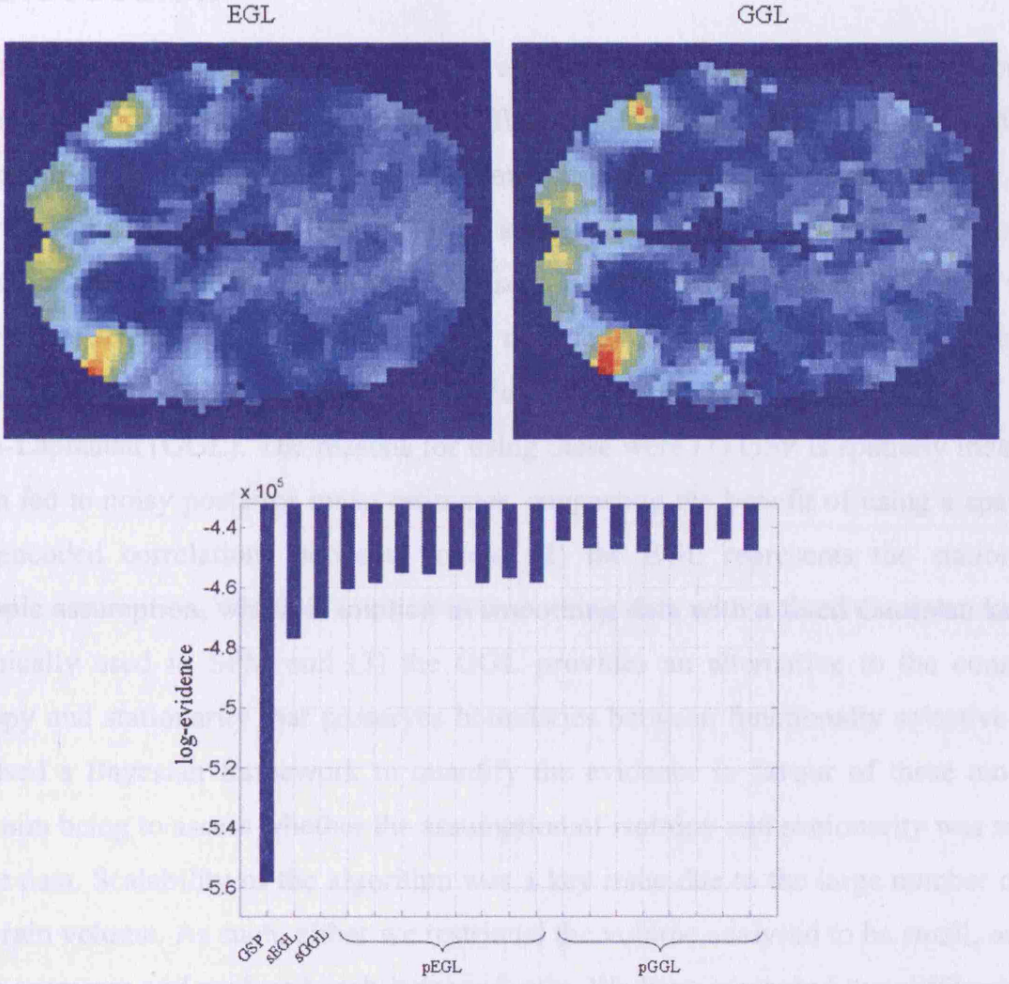


Figure 4-30: Posterior means and lower bounds (full volume; group mc)

The smallest difference in log-evidence (relative to the largest value) was between pGGL and pEGL and was $\sim 1 \times 10^4$ (Bayes factor > 100).

5 Discussion

We have outlined a Bayesian scheme to estimate the optimal smoothing of conditional parameter estimates of a GLM, using a diffusion-based spatial prior and have applied it to single-subject and group fMRI data. The main contribution is in formulating an adaptive covariance matrix in terms of the diffusion kernel on a graph, which uses ideas from image restoration and Bayesian spatial models based on GMRF and GP priors. As such there are many different forms that could have been used for this matrix, of which we have focused on three; global shrinkage prior (GSP), Euclidean graph-Laplacian (EGL) and geodesic graph-Laplacian (GGL). The reasons for using these were (1) GSP is spatially independent, which led to noisy posterior mean estimates, suggesting the benefit of using a spatial prior that encoded correlations between voxels, (2) the EGL represents the stationary and isotropic assumption, which is implicit in smoothing data with a fixed Gaussian kernel that is typically used in SPM and (3) the GGL provides an alternative to the constraint of isotropy and stationarity that preserves boundaries between functionally selective regions. We used a Bayesian framework to quantify the evidence in favour of these models, the main aim being to assess whether the assumption of isotropy and stationarity was supported by the data. Scalability of the algorithm was a key issue due to the large number of voxels in a brain volume. As such, either we restricted the volume analysed to be small, or divided it into segments and analysed each independently. We have compared two different ways to divide a volume based on (1) slices and (2) partitioning a volume using the WGL, which we compared with a model comprised of the full graph for synthetic and real data. These analyses show that a partitioned GGL model provides a parsimonious model, which is supported by the data. As the partition depends on randomly selected seed points we compared eight different partitions, which the log-evidence was on average greatest for all datasets.

There are many issues to consider in light of this work that include answering the questions; (1) is there a need for spatial models of fMRI, given that there already exists a well developed framework based on a mass-univariate approach and results from RFT to correct for multiple comparisons?, (2) what have we gained from a Bayesian perspective?

and (3) given RFs are an essential ingredient for a spatial model, what additional value is there in formulating them in terms of diffusion kernels on graphs?

An issue with the mass-univariate approach is that accurate inference over a volume requires protecting against the risk of a family-wise error (FWE), i.e. the likelihood that a family of statistical tests occurs by chance. This can be achieved by selecting an appropriate threshold based on an estimation of the smoothness of residuals from which the number of effective observations can be approximated. In conjunction with results from RFT this can be used to correct p-values and protect against FWEs, which takes into account spatial correlations in the observation error. Despite this there are a number of issues that should be considered.

Firstly, while the RF correction is principled, it was needed to correct for multiple comparisons, which was a direct consequence of analysing voxels independently. If there is one model for a family of voxels, i.e. a volume, instead of one model per voxel then inferences can be made using just one model, which does not involve multiple comparisons. This is the strategy taken here, where the posterior density over voxels is multivariate. Secondly, the estimate of smoothness is computed given the residuals *after* optimizing the GLM parameter estimates at each voxel separately. This is an example of the serial nature of the procedure, where results from one stage are passed onto another. The smoothness of the residuals depends on the degree to which data are smoothed, which in turn determines the number of RESELS and threshold chosen using the RF correction. An alternative is to have a framework where parameters of each stage are coupled through one objective function. This then provides a data driven way to optimize a spatial process (or processes for hierarchical models) i.e. a model that tries to explain data at one voxel explicitly in terms of responses of its neighbours. Thirdly, the RF correction absorbs all sources of spatial randomness into one scalar random field (the observation error). However, we may wish to explain why observations are correlated in terms of multiple sources at different levels of a hierarchy instead of taking into account their combined effect. An alternative is to represent random variability in hidden (i.e. not observed) quantities of a GLM, e.g. beta images and observation error. More importantly this leads to

the idea of how interactions between voxels could lead to the form of correlations observed and the notion of a generative model.

This brings us to the second question, whose answer involves the explicit formulation of generative models that represent randomness at each level of a hierarchy. In addition, the Bayesian framework provides an established way to formally compare different generative models through model comparison. This is useful because a generative model encodes a hypothesis about the causes of data. This is the approach taken here, where assumptions as to the form of spatial randomness are represented explicitly in the covariance of priors at each level. For the two level model used here the two random fields were over the observation error and beta values. This approach depends crucially on the theory of random fields as we are interested in representing a spatial process, i.e. one that involves interaction between different points in space. A benefit is that the random fields at each level can be different, which is useful if we have specific knowledge about randomness at a particular level. Looking ahead, the hope in generative models is that salient components of the mechanisms underlying correlations in data can be quantitatively represented in the priors.

This brings us to the third question. We know that random fields are essential because of the spatial nature of functional responses, however, these can be represented in a number of ways, e.g. GMRF or GP priors. Additional benefits of using the Laplacian to define a diffusion kernel include providing (1) established links to physical phenomena, e.g. electrical networks, heat flow and elastic media, (2) conceptual links to other established approaches, such as ReML, GMRF and GPP based schemes, (3) an adaptive basis set, where the eigensystem of a diffusion kernel is parameterized by the dispersion, τ, embedding space metric, H, and in general the expectation of the beta images, which leads to a highly adaptive basis that can be informed of spatial geometry, e.g. 2D cortical surface, and feature geometry, e.g. beta images, (4) a computationally more efficient representation, in that the Laplacian matrix is sparse compared the covariance matrices typically involved in GPPs and (5) a general framework that can be extended to include Lie groups as the "feature" at each node, e.g. DTI data (Zhang and Hancock, 2006). We consider the first two of these in more detail next.

The constitutive matrix contains the physics of a problem, e.g. stiffness of an elastic media or thermal or electrical conductivity of a material. An important point touch on above is that priors can be informed of material properties. Looking to the future, this is an advantage as it naturally incorporates intensive and extensive properties of a material, that is; intrinsic properties of the material, e.g. elasticity, and those due to its shape, or boundary conditions. Using the electrical analogy, they could also be extended to spatio-temporal random fields using RLC (resistor, inductor and capacitor) components (Bamberg and Shlomo, 1990), which could link to work on the electrical properties of neurons (Dayan and Abbott, 2001; Harrison et al., 2005), that is; the covariance matrix could be informed of possible neuronal mechanisms. In addition, we have only considered six nearest neighbour topologies here, however, these can be generalized to include long-range connections, for example small-world networks informed by results from DTI tractography.

Conceptual links to approaches based on PDEs used for image restoration, ReML, GMRF and GPP are important as they provide a resource of currently available computationally efficient techniques and potential avenues of development. There are many different forms of PDE used for image restoration, enhancement and segmentation (Aubert and Kornprobst, 2002), which could be used to generate an adaptive covariance matrix and be compared to those tested here. Formulating the model in terms of the eigenmodes of a WGL allows us to make contact with classical covariance component estimation; *i.e.,* ReML-based schemes (Friston et al., 2002b; Patterson and Thompson, 1974). This suggests using the eigenmodes as covariance components and estimating the weight of each instead of the parameterized form of weights implicit in the eigenvalues of the diffusion kernel. The link to GMRF priors suggests that the inverse diffusion kernel can be considered as an adaptive precision matrix, where the shape and scale of voxel neighbourhoods can also be optimized. In addition, techniques used in GMRF models to improved scalability, such as factorizing the posterior over voxels, can inform future developments. Links to GPPs point to continuous formulations of diffusion based spatial priors, such as using the Laplace-Beltrami operator. The advantage of this would be that they could then be used to make predictions at points in the domain that have not been measured. Next we consider details

of the algorithm, in particular, whether steps taken to increase its speed have compromised the method.

The issue of scalability is central to Bayesian spatial models. As there is only one model of the data, there is just one Laplacian, which is over all voxels in the brain. The associated spatial prior has a covariance matrix of the order $10^{5\text{-}6}$, where in lies the problem. While multiple core machines and even small clusters are becoming increasingly accessible to neuroimagers, so too is the amount and complexity of data, e.g. high-resolution fMRI. Practically this means that more powerful machines are a partial and not complete solution. A simple strategy was to restrict an analysis to a volume of interest, which is standard practice in neuroimaging. One other way to achieve this, not considered here is to first perform a standard SPM analysis using smoothed data, to produce masks of regional activity. A spatial model could then be used on non-smoothed data from this smaller volume. An alternative, which is only possible for a EGL, is to use a regular graph, i.e. the full volume including brain and non-brain voxels. This can be used in an efficient algorithm because the eigensystem of this graph Laplacian is known, i.e. eigenvectors are given by the discrete cosine set (Strang, 2007), and so does not need to be computed. However, this is very specialized in that as soon as we generalize to a WGL, where weights are no longer isotropic and stationary, the eigensystem needs to be computed.

The approach taken here was to use graphs with irregular boundaries, i.e. which excluded regions of no interest, such as outside the brain or CSF. Speed-ups were possible by (1) factorizing densities over random matrices by rows and columns and choosing a parameterized form for their covariance, (2) reducing the WGL to block-diagonal form by dividing a volume into non-overlapping segments and (3) using a fixed Laplacian, i.e. based on OLS estimates of the non-smoothed data for the GGL. We consider each of these in detail next.

Firstly the covariance at each level contains too many hyper-parameters to be estimated individually (Bishop, 2006). Matrix-variate densities provide a principled way to decompose a random matrix into rows and columns. A further massive reduction in the number of hyper-parameters comes by selecting a parameterized form of the covariance

matrix. This is also true of GPPs as they typically only involve estimating a hand full of hyper-parameters. In addition, this form can be chosen to be very simple, e.g. assuming i.i.d constraints on the measurement error. This provides a dramatic decrease in the number of quantities to be estimated, but, comes at the price of selecting an appropriate form. However, this can be turned into a model selection problem, where the goal is to find the form of covariance with the greatest evidence, which was the strategy adopted here.

Reducing the WGL to block-diagonal form was necessary to obtain segments, whose eigensystem could be easily computed using standard functions in Matlab, which also lends itself to parallel processing. This was pragmatic and not aimed at segmenting a brain volume into biologically plausible functional "objects". While some promising results were obtained using the partitioned GGL (pGGL), there was an issue in that a visible impression of partition boundaries was noticed in the posterior means. An alternative would be to simultaneously partition and estimate a brain volume using variational Bayes, based on hierarchical mixtures of experts (MoE) (Bishop, 2006), where each segment of brain data is explained by an "expert", which is a regression model for a specific region of anatomical space. Importantly the probability of a voxel being generated from an expert is learnt. This is in contrast to the current approach that effectively considers the probability of belonging to an expert as either zero or one, which leads to 'hard' decision boundaries. This means that in the current implementation we have one expert for each segment, with no mixing between them. A benefit of a MoE formulation is that the posterior density over GLM parameters is a weighted sum over experts, which will reduce boundary effects. A similar approach has been taken by (Trujillo-Barreto et al., 2004) to analyse electrophysiological data, who refer to the probability of a class label as a 'probabilistic mask'; however, these were not estimated and taken as known from an anatomical atlas. In addition, the ground node location could be included as a hyperparameter, which could be optimized similar to pseudo-inputs in (Snelson and Ghahramani, 2006). Lastly, a hierarchical MoE model could be used to optimize the number of segments as proposed in (Ueda and Ghahramani, 2002), which entails optimising the log-evidence with respect to the number of segments or mixtures.

Discussion

An alternative to the large segments used here (~1-2×10^3 voxels), would be to first reduce the dimensionality of a brain volume using anatomo-functional parcellation (Flandin et al., 2002; Thirion et al., 2006). This has been used to partition fMRI data, which employs clustering algorithms such as Gaussian mixture models (Penny and Friston, 2003) and spectral clustering (Tenenbaum et al., 2000) to divide a brain volume into many small, homogeneous regions, or parcels. This is a convenient way to reduce the dimensionality of a brain volume and has been used to perform random effects (i.e. between subjects) analysis of fMRI data. The spatial priors described here could then be applied over parcels instead of voxels.

A different strategy to segmenting a volume would be to compute the (reduced) eigensystem of the full WGL. This could be achieved using the Nystrom method (Rasmussen and Williams, 2006) or multilevel eigensolvers (Arbenz et al., 2005) based on algebraic multigrid (AMG) (Stuben, 2001). The benefit of AMG over geometric MG is that it is designed for irregular graphs instead of regular. Another promising multiscale approach is diffusion wavelets, which are an established method for fast implementation of general diffusive processes (Coifman and Maggioni, 2006; Maggioni and Mahadevan, 2006).

Gaussian process models have the same issue with scalability, which has led to developments in sparse GPPs (Lawrence, 2006; Quinonero-Candela and Rasmussen, 2005; Snelson and Ghahramani, 2007), which are used to formulate an approximate instead of a full GPP for use on large data-sets. In addition, online schemes have been used that utilize the computational efficiency of Kalman-filter like algorithms (Csato and Opper, 2002), which have been applied to large data sets from geostatistics (Cornford et al., 2005).

Using a fixed GGL based on the OLS estimate (from non-smoothed data) of beta images was pragmatic and produced some compelling results. However, preservation of high spatial frequencies or noise was apparent in some estimates using the pGGL model, which was due to noisy OLS estimates. As low order modes of the WGL capture the majority of information in a dataset, our choice of using 10% of the eigensystem to approximate the prior covariance may be inappropriate and suggests optimizing the number of eigenmodes.

Alternatively, updating the GGL and consequently the eigensystem of the prior covariance by optimizing H_f and/or using the current posterior mean estimates instead of OLS estimates could also ameliorate this. We next consider results from Chapter 4 in more detail.

The results using synthetic and real data look promising for the pGGL model with decisive evidence (Bayes factor greater than 100) for most data sets. It was useful to compare differences in the test error for synthetic data smoothed before mass-univariate OLS estimation (sOLS) with Bayesian spatial model, as it suggested the possible benefit of using an explicit spatial model, given real data. Analyses of individual slices had greatest evidence for the GGL-based model for all datasets, though this was not so for volume data. However, a common trend was for GGL to outperform (in terms of model evidence) EGL in each of the subsets of models; full, slice-wise and partitioned. That is; there does appear to be a benefit in not constraining models to be isotropic and stationary. However, of note, was that the log-evidence for the pGGL model was not greater than pEGL for all partitions for some data sets and the marked impression of partition boundaries in posterior means and preservation of noise from the OLS estimate (on which the GGL depends). These suggest that a pGGL model with soft partition boundaries and optimizing the reduced eigensystem should take priority during the next stage of development.

We end by considering future work, which will focus on more realistic noise models and application to more general spatial and feature geometries. We have considered the simplest noise model in this thesis; however, noise models with spatial extent, *i.e.* a heteroscedastic noise process, are also easily formulated using Gaussian process priors (Goldberg et al., 1998; Kersting et al., 2007; Rasmussen and Williams, 2006). A possible use in fMRI is a GPP over autoregressive model coefficients in single subject analyses following (Penny et al., 2007), who used an isotropic and stationary GMRF prior. In addition, these authors used the log-evidence computed at each voxel, thereby providing a local measure of goodness of fit, which will be included in a future implementation.

An approach, which we are currently exploring, is to generate data on, and only on, the cortical surface. This generative model could be used to explain observed responses that

have been assigned to the cortical mesh using anatomically informed basis functions (Kiebel et al., 2000). Alternatively, the model could generate 3D data by diffusing the 2D cortical response over a 3D mesh. This would have the advantage of conforming to the known anatomical generation of BOLD signal, requiring smaller prior covariance matrices, while modelling full 3D image data. An advantage of a geometric formulation of the Laplacian is that 2D coordinates of the cortical surface can be used as the anatomical space, which would lead to a non-trivial metric tensor, H_d, over physical space[17]. As the cortical mesh is constructed from an anatomical MRI a spatial prior based on such a diffusion kernel provides a way to formulate not only anatomically, but also functionally informed basis functions, thereby extending work by Kiebel *et al.*

The weights of a graph-Laplacian can be a function of scalars, vectors or matrices, which make it very flexible. For example, we have shown diffusion kernels based on distance between scalar and vector parameter fields in this thesis. However, more complex spaces, such as a field of symmetric positive definite (SPD) matrices, could be used, which would require a formulation in terms of matrix Lie groups (Rossmann, 2002). An obvious application is Diffusion Tensor Imaging (DTI) data where edge weights depend on the geodesic distance between matrices at different voxels (Zhang and Hancock, 2006). Generalizing further, Gaussian densities can be used to represent uncertainty in such matrices (Begelfor and Werman, 2005), which suggests the possibility of using a Gaussian process prior over a spatial distribution of SPD matrices, or a Lie-Gaussian process prior.

[17] Note, this could also be used for 3D analyses, where spatial distances between voxels could be used before spatial normalization instead of equally spaced nodes.

Appendices

Here we provide details of synthetic and real data sets and mathematical background in appendices I and II respectively.

I. Data sets

A. Synthetic data

A volume of data was generated containing four slices. The known parameter values of an effect of interest, design matrix and an example time-series are shown in Figure 1. The effect of interest is encoded in the first column of the design matrix, while the remaining columns contain low-frequency oscillations to simulate scanner drift and a constant term (session mean). The known spatial pattern of response is spatially non-stationary as its smoothness varies with location. The example time series is from the marked voxel and shows the temporal profile of the effect of interest (red), scanner drift (green) and observed signal (black dashed line), which includes noise. The signal-to-noise (SNR)[18] was approximately 1/10. Confounds were removed[19] for each model as described in Chapter 3. Test errors (sum of squared differences between known and estimated signal) and log-evidences for all models used to fit these data are given in Table 3 at the end of Appendix I.

B. Real data

Log-evidences for all models fitted to these data are provided in Table 4 at the end of Appendix I. These are also presented graphically in Chapter 4.

1. Auditory

---

[18] $SNR = (a_{signal} / a_{noise})^2$, where a is the root mean squared amplitude.

[19] Confounds were removed by dividing the design matrix into effects of interest, $X^{(1)}$, and confounds, $X^{(2)}$, i.e. $Y = X^{(1)}\beta^{(1)} + X^{(2)}\beta^{(2)} + \varepsilon_1$. The residual forming matrix of the confounds, $R = I - X^{(2)}(X^{(2)^T}X^{(2)})^{-1}X^{(2)^T}$ was used to adjust the data by pre-multiplying the GLM to give, $\tilde{Y} = \tilde{X}\beta + \tilde{\varepsilon}_1$ where $\tilde{Y} = RY$, $\tilde{X} = RX$, $\beta = \beta^{(1)}$ and $\tilde{\varepsilon}_1 = R\varepsilon_1$.

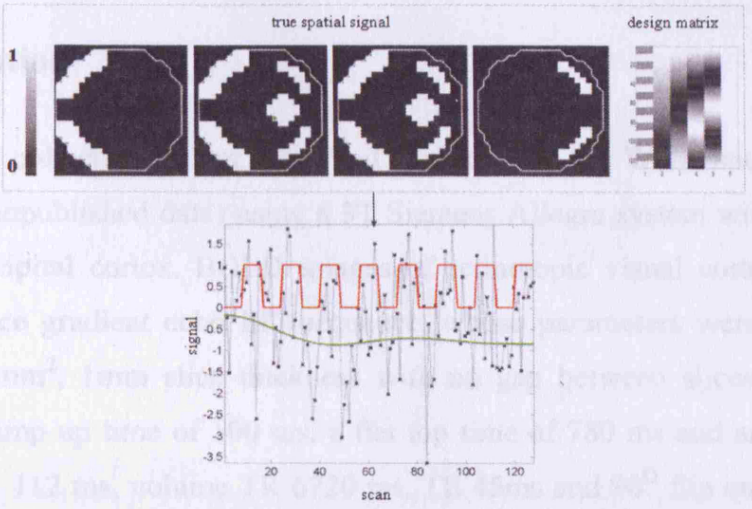


Figure 1: Known spatial signal of synthetic volume of fMRI data
The known spatial signal (four slices at the top) weighting the first column of the design matrix (top right) are shown. Each of the other columns (confounds) were weighted by a spatially independent process. Below these is shown an example time-series from the marked voxel, along with the effect of interest (red) and confounds.

This data set comprised whole brain BOLD/EPI images acquired on a modified 2T Siemens MAGNETOM Vision system. Each acquisition consisted of 64 contiguous slices (64x64x64 3mm^3 voxels). Acquisition took 6.05s, with the scan to scan repeat time (TR) set arbitrarily to 7s. 96 volumes were acquired from a single subject, in blocks of 6. The condition for successive blocks alternated between rest and auditory stimulation, starting with rest. Auditory stimulation used bi-syllabic words presented binaurally at a rate of 60 per minute. A structural image of resolution 1mm^3 was also acquired. These data are available from the SPM site http://www.fil.ion.ucl.ac.uk/spm/data/ and were pre-processed as described in the SPM manual, except for spatial smoothing. Spatial pre-processing included realignment, co-regristration, segmentation of the structural image to produce grey and white matter tissue probability maps, and normalization of the realigned functional images. This produced 46 slices of normalized functional data. Tissue probability maps of grey and white matter were normalized to the same space and voxel size as the functional images and used to produce a mask that excluded many voxels containing CSF (see "Defining a brain volume using tissue probability maps" and Figure 2).

2. High resolution

These data were collected by Drs. Ruff and Weiskopf at the Wellcome Trust centre for Neuroimaging (unpublished data) using a 3T Siemens Allegra system with a surface coil[20] centred over occipital cortex. BOLD images of retinotopic visual cortex were acquired using a multi-slice gradient echo EPI sequence, whose parameters were; 160x72 matrix, FoV = 160x72 mm^2, 1mm slice thickness with no gap between slices, trapezoidal EPI readout with a ramp up time of 100 ms, a flat top time of 780 ms and an echo spacing of 980 ms, slice TR 112 ms, volume TR 6720 ms, TE 45ms and 90^O flip angle. Each volume contained 60 contiguous slices with $1mm^2$ in-plane resolution and 1mm thickness. A total of 125 volumes were acquired, but the first 5 volumes were discarded prior to analysis to allow for T1-effects to stabilise.

The visual stimulus protocol presented standard flickering (at 10Hz) checkerboard wedge stimuli either on the horizontal or vertical meridian, each for a duration of 4 image volumes. Fifteen cycles of alternating horizontal-vertical meridian stimulation (duration 8 image volumes each) were acquired. Spatial pre-processing included realignment (Friston et al., 1995) and definition of the search volume used for subsequent GLM analyses, by means of a smoothed and thresholded brain mask image where scalp tissue voxels had been manually eroded. The design matrix used for GLM parameter estimation comprised one regressor encoding the difference between periods with vertical vs horizontal meridian stimulation, as well as several confound regressors.

3. Coherent motion

Data were collected from twelve normal subjects using a 3T Siemens Allegra system to acquire T1-weighted anatomical images and gradient-echo echo-planar T2*-weighted MRI image volumes with BOLD contrast. A total of 960 volumes were acquired per subject plus 6 initial 'dummy' volumes to allow for T1 equilibration effects. Each volume comprised 33

[20] Receive-Only 3.5cm Surface Coil (NMSC-005A, Nova Medical, Wilmington, MA, USA) for high signal-to-noise ratio in combination with a birdcage coil (NM-011 Head Transmit Coil) for radio frequency transmission. The field-of-view (FoV) was limited in the phase-encoding (PE) direction to 72 mm resulting in 72 PE lines. To avoid a (consequent) fold over artifact, we applied a saturation pulse anterior to the acquired FoV.

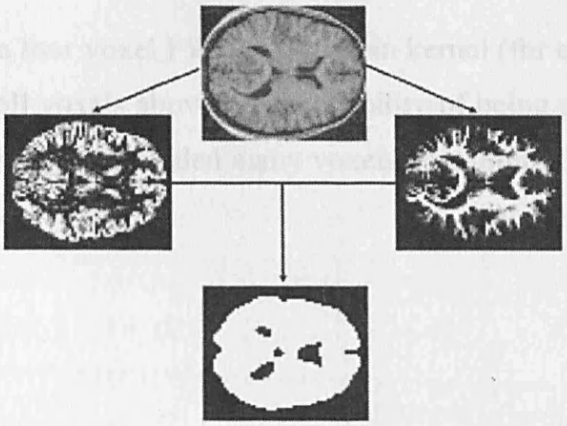


Figure 2: Using an anatomical image to define a brain volume
An anatomical image (one slice shown at the top) is segmented into grey/white matter and CSF. This produces tissue probability maps (grey and white matter maps shown left and right) that can be used to form anatomically informed masks (lower image). This mask is then used to define the spatial extent of the node set of a graph used in a spatial model of functional data.

3.3mm axial slices, with an in-plane resolution of 3×3mm, positioned to cover the entire cerebrum. Subjects were shown a visual stimulus containing an array (4×6) of circular components (each subtending 0.5°) that either oscillated about a fixed point or remained stationary. Each experiment comprised 4 sessions with a total of 1024 events, including 256 null events, when only a fixation cross was shown. Full details of the experimental paradigm can be found in (Harrison et al., 2007b). Spatial pre-processing was performed using SPM5 and included realignment, co-registration and normalization. These data were entered into a first-level (fixed effects) analysis using the standard haemodynamic response function (HRF) and its derivative as temporal basis functions. These were used to form regressors by convolution of stick functions encoding condition onsets. Condition onsets for moving and stationary stimuli were included in the model and contrast images of the effect of moving over stationary stimuli (*i.e.* using the contrast [1,-1]) were computed for both basis functions. This provided two image volumes per subject. A single subject was selected from the group for the fixed effect analysis reported in the thesis. Tissue probability maps were computed using the subjects structural MRI (resolution $1mm^3$) and a mask defined in the same way as for the auditory data set.

4. Defining a brain volume using tissue probability maps

A brain volume can be defined using tissue probability maps (Ashburner and Friston, 2005) computed from a structural image of a subject. Here we summed grey and white matter

maps, smoothed with a four voxel FWHM Gaussian kernel (for conservative coverage) and computed a mask for all voxels above 0.5 (probability of being grey or white matter). This produced a brain volume that excluded many voxels containing CSF as illustrated in Figure 2.

	sOLS	GSP	fEGL	fGGL	sEGL	sGGL
test error	91.7	50.2	89.0	**6.1**	94.0	8.8
log-evidence $\times 10^5$	-	-2.9294	-2.9235	-2.9078	-2.9240	-2.9091

		1	2	3	4	5	6	7	8
pEGL	test error	112.0	110.2	115.3	108.4	107.5	110.4	116.0	111.8
	log-evidence $\times 10^5$	-2.9272	-2.9270	-2.9274	-2.9267	-2.9264	-2.9269	-2.9277	-2.9273
pGGL	test error	6.8	8.0	9.8	12.0	8.0	7.0	11.1	8.2
	log-evidence $\times 10^5$	-2.9086	-2.9091	-2.9084	**-2.9074**	-2.9093	-2.9085	-2.9086	-2.9090

Table 3: Test error and log-evidences for all models fitted to the synthetic data set

Models with minimal test error and maximal log-evidence are shown in bold. Abbreviations for models; OLS estimate for smoothed data (sOLS), global shrinkage prior (GSP), diffusion-based spatial models applied to full volume, i.e. not divided into segments (fEGL and fGGL), applied independently to slices of a volume (sEGL and sGGL) and 3D segments using graph partitioning (pEGL and pGGL). The last two require seed points (ground nodes) to perform the segmentation, which were selected at random. This was repeated eight times to produce different partitions. The models and difference between largest and second largest (underscored) log-evidence was for pGGL-fGGL (~40; Bayes factor > 100).

data set	auditory $\times 10^4$	hi-res $\times 10^7$	single mc (reduced) $\times 10^5$	single mc (full) $\times 10^6$	group mc (reduced) $\times 10^5$	group mc (full) $\times 10^5$
GSP	-5.4032	-1.9012	3.1371	1.3949	-1.4936	-5.5819
fEGL	-5.3912	-	-	-	-	-
fGGL	-5.2074	-	-	-	-	-
sEGL	-5.3598	-1.9048	3.1591	1.4069	-1.3043	-4.7693
sGGL	-5.1865	-1.8963	3.2411	1.4309	-1.2861	-4.7009

pEGL	auditory $\times 10^4$	hi-res $\times 10^7$	single mc (restricted) $\times 10^5$	single mc (full) $\times 10^6$	group mc (restricted) $\times 10^5$	group mc (full) $\times 10^5$
1	-4.3953	-1.8998	3.1915	1.4163	-1.2497	-4.6045
2	-4.2173	-1.9002	3.2053	1.4243	-1.2700	-4.5825
3	-4.7330	-1.9000	3.2040	1.4395	-1.2523	-4.5490
4	-4.3358	-1.8997	3.1978	1.4423	-1.2673	-4.5556
5	-4.3733	-1.9002	3.2390	1.4122	-1.2419	-4.5386
6	-4.4640	-1.8992	3.1988	1.4427	-1.2432	-4.5827
7	-4.3014	-1.8994	3.2064	1.4320	-1.2497	-4.5653
8	-5.3436	-1.9000	3.1755	1.3960	-1.2510	-4.5815

pGGL	auditory $\times 10^4$	hi-res $\times 10^7$	single mc (restricted) $\times 10^5$	single mc (full) $\times 10^6$	group mc (restricted) $\times 10^5$	group mc (full) $\times 10^5$
1	-4.5971	-1.8930	3.3100	1.4599	-1.2482	-4.4427
2	-3.9645	-1.8931	3.2977	1.4345	-1.2491	-4.4665
3	**-3.9344**	**-1.8927**	3.2693	1.4474	-1.2315	-4.4707
4	-4.4359	-1.8930	3.2882	1.4374	-1.2467	-4.4790
5	-3.9872	-1.8933	3.3054	**1.4632**	-1.2458	-4.5039
6	-4.2003	-1.8932	3.2600	1.4426	-1.2286	-4.4710
7	-4.5135	-1.8931	**3.3207**	1.4360	-1.2241	**-4.4359**
8	-4.2218	-1.8933	3.3100	1.4225	**-1.2108**	-4.4737

Table 4: Log-evidences for models fitted to all real data sets

See **Table 3** for abbreviations. The models and differences in largest and second highest (underscored) log-evidence was (in order of columns); pGGL-pEGL ($\sim 3\times 10^3$), pGGL-sGGL ($\sim 3\times 10^4$), pGGL-sGGL ($\sim 8\times 10^3$), pGGL-pEGL ($\sim 2\times 10^4$), pGGL-pEGL ($\sim 3\times 10^3$) and pGGL-pEGL ($\sim 1\times 10^4$). The Bayes factor (the exponential of the number in brackets) was > 100 in all these cases.

II. Mathematical background

This appendix contains mathematical identities and results used throughout the thesis. Much of this material can be found in (Bishop, 2006; Harville, 1997; Minka, 2000)

A. Matrix identities

The matrix inversion lemma is given by

$$(A+BCD^T)^{-1} = A^{-1} - A^{-1}B(C^{-1}+D^TA^{-1}B)^{-1}D^TA^{-1} \quad \text{A.1}$$

where $A \in \Re^{n\times n}$, $B \in \Re^{n\times m}$, $C \in \Re^{m\times m}$ and $D \in \Re^{n\times m}$. A similar equation for determinants (see appendix of (Rasmussen and Williams, 2006)) is

$$\left|A+BCD^T\right| = \left|A\right|\left|C\right|\left|C^{-1}+D^TA^{-1}B\right| \quad \text{A.2}$$

Other useful identities include

$$vec(ABC) = (C^T \otimes A)vec(B) \quad \text{A.3}$$

$$\left|A\otimes B\right| = \left|A\right|^{rank(B)}\left|B\right|^{rank(A)} \quad \text{A.4}$$

$$tr(A\otimes B) = tr(A)tr(B) \quad \text{A.5}$$

$$vec(A)^T vec(B) = tr(A^TB) \quad \text{A.6}$$

$$tr(AB^T) = 1^T(A\circ B)1 \quad \text{A.7}$$

where $\circ$ is the Hadamard product and 1 is a column of ones.

$$\begin{aligned} A &= diag(a) \\ a &= diag^{-1}(A) \end{aligned} \quad \text{A.8}$$

where $A \in \Re^{n\times n}$ is diagonal with components $a \in \Re^{n\times 1}$. A useful identity is

$$diag^{-1}(AB^T) = (A \circ B)1 \qquad \text{A.9}$$

B. Eigensystem of a finite dimensional matrix

Given the real square matrix, $A \in \Re^{n \times n}$, its eigen-decomposition is

$$A = \Phi_A D_A \Phi_A^{-1} \qquad \text{B.1}$$

where $D_A = diag(\lambda_1, \lambda_1, ..., \lambda_n)$ are its eigenvalues and $\Phi_A = [\phi_1, \ldots \phi_n]$ its eigenvectors, where $\phi_i \in \Re^{n \times 1}$. If A is positive-definite [semi-definite] then $\lambda_i > 0 [\lambda_i \geq 0]$. If A is symmetric then $\Phi_A^{-1} = \Phi_A^T$. Given the eigensystem, $\{\Phi_A, D_A\}$, of a matrix, A, the matrix exponential, its inverse, determinant and trace are given by

$$\begin{aligned} \exp(A) &= \Phi_A \exp(D_A) \Phi_A^T \\ \exp(A)^{-1} &= \Phi_A \exp(-D_A) \Phi_A^T \\ \left|\exp(A)\right| &= \prod_{i=1}^{n} \exp(\lambda_i) \\ tr(\exp(A)) &= \sum_{i=1}^{n} \exp(\lambda_i) \end{aligned} \qquad \text{B.2}$$

The Kronecker product of two square matrices with eigensystems, $A = \{\Phi_A, D_A\}$ and $B = \{\Phi_B, D_B\}$, where $A \in \Re^{n \times n}$ and $B \in \Re^{m \times m}$, has the convenient form

$$A \otimes B = (\Phi_A \otimes \Phi_B)(D_A \otimes D_B)(\Phi_A \otimes \Phi_B)^{-1} \qquad \text{B.3}$$

where $A \otimes B \in \Re^{nm \times nm}$.

C. Matrix derivatives

Given a matrix A that depends on the scalar x, the derivative of its inverse and log determinant are

$$\frac{\partial}{\partial x} A^{-1} = -A^{-1} \frac{\partial A}{\partial x} A^{-1} \qquad \text{C.1}$$

$$\frac{\partial}{\partial x}\log|A| = tr\left(A^{-1}\frac{\partial A}{\partial x}\right) \qquad \text{C.2}$$

D. Gaussian densities

The matrix-variate normal (MVN) density (Gupta and Nagar, 2000) is a generalization of a univariate normal density. A univariate random variable, x, has probability density function (pdf)

$$p(x;m,s) = (2\pi s)^{-1/2}\exp(-(x-m)^2/2s) \qquad \text{D.1}$$

where $x \in \Re$, $m \in \Re$ and $s \in [0,\infty)$.

Extending this to a vector of random variables, $x = (x_1,...,x_r)^T$, where $x \in \Re^{r\times 1}$, we get the multivariate normal density,

$$p(x;m,S) = (2\pi)^{-r/2}|S|^{-1/2}\exp(-tr(S^{-1}(x-m)(x-m)^T/2) \qquad \text{D.2}$$

with mean $m \in \Re^{r\times 1}$, a $r\times r$ covariance matrix S and is represented by $x \sim N_r(m,S)$, where the sub-script represents the dimension of x.

A MVN random variable, $X \in \Re^{r\times c}$, has pdf

$$p(X;M,S,K) = (2\pi)^{-rc/2}|S|^{-c/2}|K|^{-r/2}\exp(-tr(S^{-1}(X-M)K^{-1}(X-M)^T)/2) \qquad \text{D.3}$$

with mean, $M \in \Re^{r\times c}$ and two covariance matrices, S and K, of size $r\times r$ and $c\times c$, for rows and columns respectively. This is represented by $X \sim N_{r,c}(M, S\otimes K)$, where $N_{r,c}$ stands for a MVN density (notice the row and column dimensions are separated by a comma in the sub-script). The vectorized matrix of random variables has multivariate densities $\vec{X}^T \sim N_{rc}(\vec{M}^T, S\otimes K)$ and $\vec{X} \sim N_{cr}(\vec{M}, K\otimes S)$.

Other useful results involving Gaussian densities include the following.

If $x \sim N_r(m,S)$ and $A \in \Re^{nxr}$ then $Ax \sim N_n(Am, ASA^T)$ D.4

The Kullback-Leibler (KL) relative entropy (distance) of two pdfs $q(x)$ and $p(x)$ is

$$KL(q(x) \,||\, p(x)) = \int q(x) \log\left(\frac{q(x)}{p(x)}\right) dx \geq 0 \qquad \text{D.5}$$

The KL distance for two Gaussian densities, $q = N(m_q, S_q)$ and $p = N(m_p, S_p)$ is

$$KL(q \,||\, p) = \tfrac{1}{2} \ln |\, S_p S_q^{-1} \,| + \tfrac{1}{2} tr(S_p^{-1}((m_q - m_p)(m_q - m_p)^T + S_q - S_p) \qquad \text{D.6}$$

Given the joint Gaussian density

$$\begin{bmatrix} x \\ y \end{bmatrix} \sim N\left(\begin{bmatrix} m_x \\ Am_x \end{bmatrix}, \begin{bmatrix} S_x & S_x A^T \\ AS_x & AS_x A^T + S_y \end{bmatrix}\right) \qquad \text{D.7}$$

we can use Bayes rule $p(y \mid x)p(x) = p(y)p(x \mid y)$ to give the useful result

$$N(y; Ax, S_y)N(x; m_x, S_x) = N(y; Am_x, S)N(x; m_{x|y}, S_{x|y})$$

$$\begin{aligned} S &= AS_x A^T + S_y \\ m_{x|y} &= m_x + C(y - Am_x) \\ S_{x|y} &= (S_x^{-1} + A^T S_y^{-1} A)^{-1} \end{aligned} \qquad \text{D.8}$$

$$C = S_{x|y} A^T S_y^{-1}$$

Alternative expressions commonly seen are $C = S_x A^T S^{-1}$ and $S_{x|y} = (I - CA)S_x$. These expression relate to those in Eqn 3.11 using $S_y \to \Sigma_1$, $S_x \to \Sigma_2$, $A \to Z$, $m_x \to 0$, $x \to b$, $m_{x|y} \to \overline{b}$ and $S_{x|y} \to \Pi$, so $m_{x|y} = Cy \to \overline{b} = \Pi Z^T \Sigma_1^{-1} y$.

The expectation of a quadratic form is

$$\int dx (y - Ax)^T \Sigma^{-1} (y - Ax) N(m,S) = (y - Am)^T \Sigma^{-1} (y - Am) + tr(A^T \Sigma^{-1} AS) \qquad \text{D.9}$$

"Completing the square" is used to re-arrange the quadratic form in the exponent of a Gaussian density

$$-\frac{1}{2}(x-m)^T S^{-1}(x-m) = -\frac{1}{2}x^T S^{-1}x + xS^{-1}m + const \qquad \text{D.10}$$

E. Approximate model evidence

Given a generative model, m, with data, y, hidden variables, b and hyper-parameters, α, the joint distribution over data and hidden variables $p(y,b \mid \alpha)$, decomposes according to Bayes rule

$$p(y,b \mid \alpha) = p(y \mid b,\alpha)p(b \mid \alpha) = p(b \mid y,\alpha)p(y \mid \alpha) \qquad \text{E.1}$$

where the likelihood, prior, true posterior and marginal likelihood are $p(y \mid b,\alpha)$, $p(b \mid \alpha)$, $p(b \mid y,\alpha)$ and $p(y \mid \alpha)$ respectively. For Gaussian densities the latter two are given in D.8, where the marginal likelihood is a normalization term, *i.e.* $p(y \mid \alpha) = \int_b p(y \mid b,\alpha)p(b \mid \alpha)$, which can be used to approximate the model evidence, *i.e.* the probability of the data given the model [21]. It is this quantity that we wish to optimize with respect to the free hyper-parameters of the model.

The log marginal likelihood can be written, by re-arranging E.1 and taking logs, as the log ratio of joint and true posterior densities

$$\log p(y|\alpha) = \log\left(\frac{p(y,b \mid \alpha)}{p(b \mid y,\alpha)}\right) \qquad \text{E.2}$$

[21] Technically this is the evidence of the hyper-parameters. The model evidence is obtained by integrating out the hyper-parameters, *i.e.* $p(y \mid m) = \int_\alpha p(y \mid \alpha,m)p(\alpha \mid m)$. We approximate this using Laplace's method by expanding about the final estimate of α.

This can be approximated by a lower bound by specifying an approximate posterior over the hidden variables, $q(b)$. Integration of E.2 over b then allows the log marginal likelihood to be decomposed into two terms

$$\begin{aligned}\log p(y|\alpha) &= \int_b q(b)\log\left(\frac{p(y,b\mid\alpha)}{p(b\mid y,\alpha)}\right) = \int_b q(b)\log\left(\frac{p(y,b\mid\alpha)}{p(b\mid y,\alpha)}\frac{q(b)}{q(b)}\right)\\ &= \int_b q(b)\log\left(\frac{p(y,b\mid\alpha)}{q(b)}\right) + \int_b q(b)\log\left(\frac{q(b)}{p(b\mid y,\alpha)}\right)\\ &= F(q(b),\alpha) + KL(q(b)\,||\,p(b\mid y,\alpha)\end{aligned} \qquad \text{E.3}$$

Where we have used the Kullback-Leibler relative entropy (see D.5). We can see from this that $F(q,\alpha)$ is a lower bound to $\log p(y\mid\alpha)$, because $KL(q(b)\,||\,p(b\mid y,\alpha))\geq 0$, which implies that $\log p(y\mid\alpha)\geq F(q,\alpha)$. This lower bound can be re-written in terms of quantities which we have access to, *i.e.* the likelihood, prior and approximate posterior density over hidden variables, $p(y\mid b,\alpha)$, $p(b)$ and $q(b)$ respectively.

$$\begin{aligned}F(q,\alpha) &= \int q(b)\log\left(\frac{p(y,b\mid\alpha)}{q(b)}\right)db = \int q(b)\log\left(\frac{p(y\mid b,\alpha)p(b)}{q(b)}\right)db\\ &= \int q(b)\log p(y\mid b,\alpha)db - \int q(b)\log\left(\frac{q(b)}{p(b)}\right)db\\ &= \left\langle\log p(y\mid b,\alpha)\right\rangle_{q(b)} - KL(q(b)\,||\,p(b))\end{aligned} \qquad \text{E.4}$$

Optimizing the log marginal likelihood, $\log p(y\mid\alpha)$, can then be cast as a lower bound, $F(q,\alpha)$, optimization, which can be solved using gradient ascent. This is the objective function shown in Chapter 3.

An alternative formulation of the lower bound used in the main text, *e.g.* Eqn 3.3, is achieved using D.9 and D.6 with expressions for the likelihood, prior and posterior over parameters; $p(y\mid b)=N(y;Zb,\Sigma_1)$, $p(b)=N(b;0,\Sigma_2)$, and $q(b)=N(b;\overline{b},\Pi^{-1})$ respectively. The expected likelihood and KL distance between the posterior and prior are then given by

$$\begin{aligned}\left\langle \log p(y \mid b,\alpha)\right\rangle_{q(b)} &= \int db (y - Zb)^T \Sigma_1^{-1} (y - Zb) N(\bar{b}, \Pi^{-1}) \\ &= (y - Z\bar{b})^T \Sigma_1^{-1} (y - Z\bar{b}) + tr(Z^T \Sigma_1^{-1} Z \Pi^{-1}) \\ KL(q(b) \| p(b)) &= \tfrac{1}{2} \ln |\Sigma_2 \Pi| + \tfrac{1}{2} tr(\Sigma_2^{-1} (\bar{b}\bar{b}^T + \Pi^{-1} - \Sigma_2))\end{aligned} \quad \text{E.5}$$

Writing out each term explicitly, the lower bound is

$$\begin{aligned}F(q,\alpha) &= -\tfrac{1}{2}(\ln|\Sigma_1| + \bar{e}_1^T \Sigma_1^{-1} \bar{e}_1 + tr(Z^T \Sigma_1^{-1} Z \Pi^{-1})) - \\ &\quad \tfrac{1}{2}(\ln|\Sigma_2 \Pi| + \bar{b}^T \Sigma_2^{-1} \bar{b} + tr(\Sigma_2^{-1} \Pi^{-1}) - tr(I_{PN})) \\ &= -\tfrac{1}{2}(\ln|\Sigma_1| + \ln|\Sigma_2| + \ln|\Pi| + \bar{e}_1^T \Sigma_1^{-1} \bar{e}_1 + \bar{b}^T \Sigma_2^{-1} \bar{b} + TN \log 2\pi)\end{aligned} \quad \text{E.6}$$

Given that the total covariance, Σ, is comprised of two parts, Σ_1 and Σ_2, we can complete the square (D.10) to get

$$y^T \Sigma^{-1} y = (y - Z\bar{b})^T \Sigma_1^{-1} (y - Z\bar{b}) + \bar{b}^T \Sigma_2^{-1} \bar{b} \quad \text{E.7}$$

And using the log of A.2, the lower bound is equivalent to

$$\begin{aligned}F(q,\alpha) &= -\tfrac{1}{2}\left(\ln|\Sigma(\alpha)| + y^T \Sigma(\alpha)^{-1} y + TN \ln 2\pi\right) \\ \Sigma(\alpha) &= \Sigma_1 + Z \Sigma_2 Z^T\end{aligned} \quad \text{E.8}$$

Which is the same as that in Eqn 3.3.

In practice, optimization of non-negative scale parameters in the **M**-Step uses the transformation; $\gamma_i = \ln \alpha_i$. The derivatives in Table 5 are then $\partial K / \partial \gamma_i = \alpha_i \, \partial K / \partial \alpha_i$. Under this change of variables, the hyper-parameters have non-informative log-normal hyper-priors. Uncertainty about the hyper-parameters can be included in the log-evidence for a model *m*. For example, the approximate log-evidence including uncertainty of one hyper-parameter is

$$\begin{aligned}\ln p(y|m) &\approx \ln p(y \mid \gamma, m) - \tfrac{1}{2}\ln\frac{\partial^2 F}{\partial \gamma^2} \\ &\approx \ln p(y|\gamma, m) + \tfrac{1}{2}\ln I \\ \frac{\partial^2 F}{\partial \gamma^2} &\approx \left\langle \frac{\partial^2 F}{\partial \gamma^2} \right\rangle\end{aligned} \qquad \text{E.9}$$

Where we have approximated the second-order derivative (see F.4) using the expected information computed in the **M**-step. See (Friston et al., 2007) for details.

F. Fisher-scoring scheme

Given the log-marginal likelihood (see E.8)

$$\ln p(y \mid \alpha) \geq F = -\tfrac{1}{2}\left(\ln|\Sigma| + y^T \Sigma^{-1} y + TN \ln 2\pi\right) \qquad \text{F.1}$$

The hyper-parameters (indexed by subscripts) can be updated, *i.e.* $\alpha^{(new)} = \alpha^{(old)} + \Delta\alpha$, using a Fisher-scoring scheme[22], where

$$\Delta\alpha = I(\alpha)^{-1} \nabla_\alpha F \qquad \text{F.2}$$

$\nabla_\alpha F$ is the score, *i.e.* a vector of gradients (k^{th} element given by $\partial F / \partial \alpha_k$) with respect to covariance hyper-parameters

$$\frac{\partial F}{\partial \alpha_k} = -\tfrac{1}{2} tr\left(\Sigma^{-1}\frac{\partial \Sigma}{\partial \alpha_k}\right) + \tfrac{1}{2} y^T \Sigma^{-1} \frac{\partial \Sigma}{\partial \alpha_k} \Sigma^{-1} y \qquad \text{F.3}$$

where Σ is the current maximum likelihood estimate of the data covariance. The expected information matrix, $I(\alpha)$, see (Wand, 2002), with element I_{kl}, is the negative of the expectation (over the marginal likelihood of the data) of the second derivative. This latter quantity is

[22] This is equivalent to a Newton step, but using the expected curvature as opposed to the local curvature of the objective function.

$$\frac{\partial^2 F}{\partial\alpha_k\partial\alpha_l} = \tfrac{1}{2}tr\left(\Sigma^{-1}\frac{\partial\Sigma}{\partial\alpha_k}\Sigma^{-1}\frac{\partial\Sigma}{\partial\alpha_l}\right) - y^T\Sigma^{-1}\frac{\partial\Sigma}{\partial\alpha_k}\Sigma^{-1}\frac{\partial\Sigma}{\partial\alpha_l}\Sigma^{-1}y \qquad \text{F.4}$$

The expectation with respect to $p(y\,|\,\alpha)$, in F.4, does not change the first term and the second, using D.9, is

$$\int_y y^T\Sigma^{-1}\frac{\partial\Sigma}{\partial\alpha_k}\Sigma^{-1}\frac{\partial\Sigma}{\partial\alpha_l}\Sigma^{-1}yN(0,\Sigma) = tr\left(\Sigma^{-1}\frac{\partial\Sigma}{\partial\alpha_k}\Sigma^{-1}\frac{\partial\Sigma}{\partial\alpha_l}\right) \qquad \text{F.5}$$

The expected Information is then given by

$$I_{kl} = -\left\langle\frac{\partial^2 F}{\partial\alpha_k\partial\alpha_l}\right\rangle = \tfrac{1}{2}tr\left(\Sigma^{-1}\frac{\partial\Sigma}{\partial\alpha_k}\Sigma^{-1}\frac{\partial\Sigma}{\partial\alpha_l}\right) \qquad \text{F.6}$$

G. Linear algebra for the EM scheme

Here we provide notes on the linear algebra used to compute the gradients and curvatures necessary for the **EM** scheme in the main text. They are not necessary to understand the results presented above but help optimise implementation.

We require the bound on the log-marginal likelihood, $\ln p(y\,|\,\alpha)$ and its derivatives.

$$\begin{aligned} F &= -\tfrac{1}{2}\left(\ln|\Sigma| + y^T\Sigma^{-1}y\right) + const \\ \Sigma(\alpha) &= \Sigma_1 + Z\Sigma_2 Z^T \\ \Sigma_i &= K_i \otimes S_i \end{aligned} \qquad \text{G.1}$$

Using A.2 the first term of G.1 is

$$\begin{aligned} \ln|\Sigma| &= \ln|\Sigma_1| + \ln|\Sigma_2| + \ln|\Sigma_2^{-1} + Z^T\Sigma_1^{-1}Z| \\ &= \ln|\Sigma_1| + \ln|\Sigma_2| + \ln|\Pi| \end{aligned} \qquad \text{G.2}$$

and this can be reduced further using A.4. The second term is

$$y^T\Sigma^{-1}y = tr(Y^T A_{\varepsilon_1})$$
$$A_{\varepsilon_1} = S_1^{-1}\bar{\varepsilon}_1 K_1^{-1} \qquad \text{G.4}$$

where we have used A.6 and $\bar{\varepsilon}_1 = Y - X\bar{\beta}$ is the matrix of prediction errors, where $\bar{b} = vec(\bar{\beta})$.

*Conditional moments of parameters (**E**-step)*

The conditional precision is (see D.8)

$$\Pi = Z^T\Sigma_1^{-1}Z + \Sigma_2^{-1} = K_1^{-1} \otimes X^T S_1^{-1} X + K_2^{-1} \otimes S_2^{-1} \qquad \text{G.5}$$

The conditional covariance can be formulated in terms of eigenmodes (B.1) of the second level prior covariance as follows: using the matrix inversion lemma (A.1) the data precision is

$$\begin{aligned}\Sigma^{-1} &= \Sigma_1^{-1} - \Sigma_1^{-1}Z(\Sigma_2^{-1} + Z^T\Sigma_1^{-1}Z)^{-1}Z^T\Sigma_1^{-1} \\ &= \Sigma_1^{-1} - \Sigma_1^{-1}Z\Pi^{-1}Z^T\Sigma_1^{-1}\end{aligned} \qquad \text{G.6}$$

Using the eigenvalue decomposition; $\Sigma_2 = \Phi_2 D_2 \Phi_2^T$, and B.3, where $\Phi_2 = \Phi_{K_2} \otimes \Phi_{S_2}$, $D_2 = D_{K_2} \otimes D_{S_2}$, then

$$\begin{aligned}\Sigma &= Z\Phi_2 D_2 \Phi_2^T Z^T + \Sigma_1 \\ \Sigma^{-1} &= \Sigma_1^{-1} - \Sigma_1^{-1}Z\Phi_2(D_2^{-1} + \Phi_2^T Z^T\Sigma_1^{-1}Z\Phi_2^T)^{-1}\Phi_2^T Z^T\Sigma_1^{-1}\end{aligned} \qquad \text{G.7}$$

Comparing the last lines of G.6 with G.7

$$\begin{aligned}\Pi^{-1} &= \Phi_2 E \Phi_2^T \\ E &= (D_2^{-1} + \Phi_2^T Z^T\Sigma_1^{-1}Z\Phi_2^T)^{-1} \\ &= D_2^{1/2}(I + D_2^{1/2}\Phi_2^T Z^T\Sigma_1^{-1}Z\Phi_2^T D_2^{1/2})^{-1} D_2^{1/2}\end{aligned} \qquad \text{G.8}$$

Note, for a diffusion-based prior $D_{K_2} = g(\Lambda) = \exp(-\Lambda\tau)$, however, we could use $D_{K_2} = g(\Lambda) = \Lambda^{-1}$ for a Laplacian prior (numerically stable expressions for each are given

in the last and penultimate lines of G.8 respectively). Note also that the symbol E, in Eqn G.8, does not represent the edge set in this case.

The conditional mean is (see D.8)

$$\overline{b} = \Phi_2 E \Phi_2^T Z^T \Sigma_1^{-1} y \qquad \text{G.9}$$

*Conditional moments of hyper-parameters (**M**-step)*

To compute the derivatives required for the **M**-step, we use standard results for Kronecker tensor products to show the score and expected information reduce to

$$\frac{\partial F}{\partial \alpha_k} = -\tfrac{1}{2} tr\left(A_a^{(k)} \otimes B_a^{(k)} - (F_a^{(k)} C \otimes G_a^{(k)} D) E + A_\varepsilon^T \widetilde{B}_a^{(k)} A_\varepsilon \widetilde{A}_a^{(k)T} \right) \qquad \text{G.10}$$

and

$$\frac{\partial^2 I}{\partial \alpha_k \partial \alpha_l} = \tfrac{1}{2} tr(A_a^{(k)} A_b^{(l)}) tr(B_a^{(k)} B_b^{(l)}) + \ldots$$

$$\tfrac{1}{2} tr\left((F_b^{(l)} C \otimes G_b^{(l)} D) E (F_a^{(k)} C \otimes G_a^{(k)} D) E \right) - \ldots \qquad \text{G.11}$$

$$\tfrac{1}{2} tr\left((F_b^{(l)} A_a^{(k)} C \otimes G_b^{(l)} B_a^{(k)} D) E \right) - \tfrac{1}{2} tr\left((F_a^{(k)} A_b^{(l)} C \otimes G_a^{(k)} B_b^{(l)} D) E \right)$$

where the superscript of matrices A, B, F, G represents a hyper-parameter index, *i.e.* $k, l \in \{1,2,3\}$, while the subscript represents a level index for error covariances, *i.e.* $a, b \in \{1,2\}$, which will simplify expressions later. Terms in G.10 and 11 are given by

$$
\begin{aligned}
\widetilde{A}_a^{(k)} \otimes \widetilde{B}_a^{(k)} &= \frac{\partial \Sigma}{\partial \lambda_k} \\
A_a^{(k)} &= K_1^{-1} \widetilde{A}_a^{(k)} \\
B_a^{(k)} &= S_1^{-1} \widetilde{B}_a^{(k)} \\
C &= K_1^{-1} \Phi_{K_2} \\
D &= S_1^{-1} X \Phi_{S_2} \\
E &= D_2^{1/2} (I + D_2^{1/2} \Phi_2^T Z^T \Sigma_1^{-1} Z \Phi_2 D_2^{1/2})^{-1} D_2^{1/2} \\
F_a^{(k)} &= \Phi_{K_2}^T K_1^{-1} \widetilde{A}_a^{(k)} \\
G_a^{(k)} &= \Phi_{S_2}^T X^T S_1^{-1} \widetilde{B}_a^{(k)}
\end{aligned}
\qquad \text{G.12}
$$

Supporting calculations for the score, using the matrix derivatives in C.1 and 2, are

$$
\frac{\partial F}{\partial \alpha_k} = -\tfrac{1}{2}\left(tr(\Sigma^{-1} \frac{\partial \Sigma}{\partial \alpha_k}) + y^T \Sigma^{-1} \frac{\partial \Sigma}{\partial \alpha_k} \Sigma^{-1} y \right)
$$

$$
\begin{aligned}
tr(\Sigma^{-1} \frac{\partial \Sigma}{\partial \alpha_k}) &= tr\left((\Sigma_1^{-1} - \Sigma_1^{-1} Z \Pi^{-1} Z^T \Sigma_1^{-1}) \widetilde{A}_a^{(k)} \otimes \widetilde{B}_a^{(k)} \right) \\
&= tr\left(A_a^{(k)} \otimes B_a^{(k)} - (C \otimes D) E (F_a^{(k)} \otimes G_a^{(k)}) \right) \\
&= tr\left(A_a^{(k)} \otimes B_a^{(k)} - (F_a^{(k)} C \otimes G_a^{(k)} D) E \right)
\end{aligned}
\qquad \text{G.13}
$$

and

$$
\begin{aligned}
y^T \Sigma^{-1} \frac{\partial \Sigma}{\partial \alpha_k} \Sigma^{-1} y &= vec(A_\varepsilon)^T (\widetilde{A}_a^{(k)} \otimes \widetilde{B}_a^{(k)}) vec(A_\varepsilon) \\
&= vec(A_\varepsilon)^T vec(\widetilde{B}_a^{(k)} A_\varepsilon \widetilde{A}_a^{(k)^T}) \\
&= tr(A_\varepsilon^T \widetilde{B}_a^{(k)} A_\varepsilon \widetilde{A}_a^{(k)^T})
\end{aligned}
\qquad \text{G.14}
$$

where we have used $\Pi^{-1} = \Phi_2 E \Phi_2^T$ and the notation in G.12. The expression in G.11 is derived from the expected Fisher Information, $I_{kl} = -\left\langle \partial^2 F / \partial \alpha_k \partial \alpha_l \right\rangle$, using F.6, G.13 and the cyclic property of trace. These expressions simplify further using A.5. Note, if the data are transformed, *i.e.* $\widetilde{Y} = P_r Y P_c$, then all variables are transformed as shown in Eqn 3.8.

k	1	2	3
Hyper-parameter	υ	τ	η
$\frac{\partial \Sigma}{\partial \gamma_k}$	$\frac{\partial K_1}{\partial \gamma_1} \otimes S_1$	$\frac{\partial K_2}{\partial \gamma_2} \otimes XS_2X^T$	$K_2 \otimes X\frac{\partial S_2}{\partial \gamma_3}X^T$
$\widetilde{A}_a^{(k)}$	K_1	$-LK_2\tau$	K_2
$\widetilde{B}_a^{(k)}$	S_1	XS_2X^T	XS_2X^T

Table 5: Derivatives of data covariance matrix

$\overline{K}_{a1b}^{-1} = \Phi_{K_a}^T K_1^{-1} \Phi_{K_b}$	1	2
1	$\Phi_{K_1}^T K_1^{-1} \Phi_{K_1}$	$\Phi_{K_1}^T K_1^{-1} \Phi_{K_2}$
2	$\Phi_{K_2}^T K_1^{-1} \Phi_{K_1}$	$\Phi_{K_2}^T K_1^{-1} \Phi_{K_2}$

Table 6: Column precisions

$\overline{S}_{a1b}^{-1}$	1	2
1	$\Phi_{S_1}^T S_1^{-1} \Phi_{S_1}$	$\Phi_{S_1}^T S_1^{-1} X \Phi_{S_2}$
2	$\Phi_{S_2}^T X^T S_1^{-1} \Phi_{S_1}$	$\Phi_{S_2}^T X^T S_1^{-1} X \Phi_{S_2}$

Table 7: Row precisions

	1	2	3
$dD_a^{(k)}$	$D_{K_1} \otimes D_{S_1}$	$-\Lambda_2 D_{K_2} \tau_2 \otimes D_{S_2}$	$D_{K_2} \otimes D_{S_2}$

Table 8: Eigenvalues of derivatives

The formulation above is not a computationally efficient way to implement the algorithm. We want to make use of $K_i = \Phi_{K_i} D_{K_i} \Phi_{K_i}^T$ and $S_i = \Phi_{S_i} D_{S_i} \Phi_{S_i}^T$, in particular, given $L = \Phi_{K_2} \Lambda \Phi_{K_2}^T$, $\exp(-L\tau) = \Phi_{K_2} D_{K_2} \Phi_{K_2}^T$ and $D_{K_2} = g(\Lambda, \tau) = \exp(-\Lambda\tau)$

Computationally efficient expressions are obtained using A.7 and the following

$$
\begin{aligned}
tr(\overline{A}_a^{(k)}) &= tr(A_a^{(k)}) \\
tr(\overline{B}_a^{(k)}) &= tr(B_a^{(k)}) \\
\overline{C}_a^{(k)} &= F_a^{(k)} C \\
\overline{D}_a^{(k)} &= G_a^{(k)} D \\
\overline{F}_{ab}^{(kl)} &= F_a^{(k)} A_b^{(l)} C \\
\overline{G}_{ab}^{(kl)} &= G_a^{(k)} B_b^{(l)} D
\end{aligned}
\qquad \text{G.15}
$$

together with the expressions in Table 6, Table 7 and Table 8. Here we have used the notation $\overline{X}_{alb} = \Phi_a^T X_1 \Phi_b$ to represent left and right multiplication of X_1 by bases Φ_a and Φ_b respectively. This is important when reducing the number of eigenmodes, *e.g.* $n_a, n_b < N$, as the dimension of X_1 is reduced from $N \times N$ to $n_a \times n_b$. Components of G.10 and 11 then can be written

$$\begin{aligned} tr(A_a^{(k)}) &= tr(\overline{A}_a^{(k)}) \\ &= 1^T(\overline{K}_{a1a}^{-1} \circ dD_a^{(k)})1 \\ tr(B_a^{(k)}) &= tr(\overline{B}_a^{(k)}) \\ &= 1^T(\overline{S}_{a1a}^{-1} \circ dD_a^{(k)})1 \end{aligned} \qquad \text{G.16}$$

$$\begin{aligned} tr((F_a^{(k)}C \otimes G_a^{(k)}D)E) &= tr((\overline{C}_a^{(k)} \otimes \overline{D}_a^{(k)})E) \\ &= 1^T((\overline{C}_a^{(k)} \otimes \overline{D}_a^{(k)}) \circ E^T)1 \end{aligned} \qquad \text{G.17}$$

$$\begin{aligned} tr(A_a^{(k)}A_b^{(l)}) &= 1^T((\overline{K}_{b1a}^{-1} dD_a^{(k)}) \circ (\overline{K}_{a1b}^{-1} dD_b^{(l)})^T)1 \\ tr(B_a^{(k)}B_b^{(l)}) &= 1^T((\overline{S}_{b1a}^{-1} dD_a^{(k)}) \circ (\overline{S}_{a1b}^{-1} dD_b^{(l)})^T)1 \end{aligned} \qquad \text{G.18}$$

$$\begin{aligned} &tr((F_a^{(k)}C \otimes G_a^{(k)}D)E(F_b^{(l)}C \otimes G_b^{(l)}D)E) \\ &= tr((\overline{C}_a^{(k)} \otimes \overline{D}_a^{(k)})E(\overline{C}_b^{(l)} \otimes \overline{D}_b^{(l)})E) \\ &= 1^T((\overline{C}_a^{(k)} \otimes \overline{D}_a^{(k)})E \circ ((\overline{C}_b^{(l)} \otimes \overline{D}_b^{(l)})E)^T)1 \end{aligned} \qquad \text{G.19}$$

$$\begin{aligned} tr((F_a^{(k)}A_b^{(l)}C \otimes G_a^{(k)}B_b^{(l)}D)E &= tr((\overline{F}_{ab}^{(kl)} \otimes \overline{G}_{ab}^{(kl)})E) \\ &= 1^T((\overline{F}_{ab}^{(kl)} \otimes \overline{G}_{ab}^{(kl)}) \circ E^T)1 \end{aligned} \qquad \text{G.20}$$

The expressions for $tr(A_a^{(k)})$ and $tr(B_a^{(k)})$ are sparse because $dD_a^{(k)}$ is diagonal, even if $\overline{K}_{a1a}^{-1}$ or $\overline{S}_{a1a}^{-1}$ are not.

H. Computing posterior probability maps

A posterior probability map has two thresholds $t_1 \in \Re$ and $t_2 \in [0,1]$ that are used to show voxels were the model is at least $100 \times t_2\%$ certain that the effect size is greater than t_1 and

is represented by the expression $p(u > t_1) > t_2$, where $u = C^T b$ is a contrast, *i.e.* linear combination of GLM parameters. Given a contrast vector c of size $P \times 1$, where $C = I_{N_v} \otimes c$, the posterior density over u is

$$q(u) = N_{N_V}(\overline{u}, C^T \Pi^{-1} C) \qquad \text{H.1}$$

This is used to produce a statistical image by considering the diagonal components of its covariance, $\upsilon = diag^{-1}(C^T \Pi^{-1} C)$, which is computed using A.9 and G.8

$$\upsilon = ((C^T \Phi E) \circ (C^T \Phi))1 \qquad \text{H.2}$$

The probability, at the i^{th} voxel, of u being above a threshold t_1 is then

$$p_i = 1 - \int_{-\infty}^{t_1} q(u_i) \qquad \text{H.3}$$

where $q(u_i) = N(\overline{u}_i, \upsilon_i)$. A binary image is then computed using the threshold, t_2

$$P_i = \begin{cases} 1 & if \quad p_i > t_2 \\ 0 & otherwise \end{cases} \qquad \text{H.4}$$

where P is the PPM over the node set V.

I. Metrics on manifolds

The intuition behind the induced metric comes from considering Pythagoras' theorem in two dimensions.

$$ds^2 = du^2 + hdf^2 = \left(1 + h\left(\frac{df}{du}\right)^2\right) du^2 = \widetilde{G} du^2 \qquad \text{I.1}$$

More formally, consider a one-dimensional curve embedded in two-dimensional Euclidean space. A map from one manifold, (M,g), to another, (N,h), where G and H are metrics associated with each respectively, is

$$\begin{aligned}&\chi : M \to N \\ &\chi : u \to \left(\chi^1(u), \chi^2(u)\right) = \left(u, f(u)\right)\end{aligned} \qquad \text{I.2}$$

Where u is a local coordinate on the curve and χ^1 and χ^2 are coordinates in the embedding space. A distance ds on the curve in terms of du is given by

$$\begin{aligned}H &= \begin{pmatrix}1 & 0\\ 0 & h\end{pmatrix} \\ J &= \frac{\partial \chi}{\partial u} = \begin{pmatrix}1\\ f_u\end{pmatrix} \\ G &= J^T H J = \begin{pmatrix}1 & f_u\end{pmatrix}\begin{pmatrix}1 & 0\\ 0 & h\end{pmatrix}\begin{pmatrix}1\\ f_u\end{pmatrix} \\ &= 1 + h f_u^2 \\ ds^2 &= G du^2\end{aligned} \qquad \text{I.3}$$

Where the relative scale between the domain and feature coordinates is h and G is the induced metric *i.e.* metric on the curve, and $G \approx \widetilde{G}$.

J. Computing the graph Laplacian

We assemble the 3D graph Laplacian using a 6 nearest neighbours. In this thesis the matrix that scales feature displacement (see Eqn 2.11) is chosen to be

$$\begin{aligned}H_f &= C_{ols}^{-1} \\ C_{ols} &= (\beta_{ols} - M_{ols})(\beta_{ols} - M_{ols})^T \\ M_{ols} &= \frac{1}{N_V}\beta_{ols} 1_{N_V} 1_{N_V}^T\end{aligned} \qquad \text{J.1}$$

where β_{ols} is the OLS estimate of GLM parameters (given non-smoothed data) and 1_{N_V} is a column vector of ones of length N_V.

K. Updating the graph-Laplacian

Generally, during optimization, the Laplacian is a function of GLM parameters, $L(f)$, which are unknown and therefore have a degree of uncertainty. The matrix P_{dt} (see below and Eqn 2.33) can be approximated by expanding around the current posterior mean as follows, where we use the same notation as in subsection 2.4, i.e. $f = Zb$

$$\frac{df}{dt} = -\tfrac{1}{2} L(f) f \Rightarrow f_{t+dt} = P_{dt} f_t$$

$$\begin{aligned} f_{t+dt} &\approx h(f_t)\big|_{f_t=\bar{f}_t} + \left.\frac{\partial h}{\partial f}\right|_{f_t=\bar{f}_t} (f_t - \bar{f}_t) \\ &\approx \exp(-L(\bar{f}_t)dt) f_t \end{aligned} \qquad \text{K.1}$$

where we have used

$$\begin{aligned} P_{dt} &= \exp(-L(f_t)dt) \\ h(f_t) &= P_{dt} f_t \\ \left.\frac{\partial h}{\partial f}\right|_{f_t=\bar{f}_t} &\approx \exp(-L(\bar{f}_t)dt) \\ h(f_t)\big|_{f_t=\bar{f}_t} &= \exp(-L(\bar{f}_t)dt)\bar{f}_t \end{aligned} \qquad \text{K.2}$$

In this thesis we do not update the Laplacian during optimization. Instead it is fixed using the OLS estimates of non-smoothed data, *i.e.* $L(f_{ols})$. This means that we use $f_{t+dt} \approx \exp(-L(f_{ols})dt) f_t$ instead of the last line in K.1.

Bibliography

Abrahamsen, P. (1997). A review of Gaussian random fields and correlation functions (Oslo, Norwegian Computing Centre).

Adler, R. J. (1981). The Geometry of Random Fields (London, Wiley).

Adler, R. J., and Taylor, J. (2007). Random fields and geometry (New York, Springer-Verlag).

Alvarez, L., Lions, P. L., and Morel, J. M. (1992). Image Selective Smoothing and Edge-Detection by Nonlinear Diffusion.2. Siam Journal on Numerical Analysis *29*, 845-866.

Andrade, A., Kherif, F., Mangin, J. F., Worsley, K. J., Paradis, A. L., Simon, O., Dehaene, S., Le Bihan, D., and Poline, J. B. (2001). Detection of fMRI activation using cortical surface mapping. Hum Brain Mapp *12*, 79-93.

Arbenz, P., Hetmaniuk, U. L., Lehoucq, R. B., and Tuminaro, R. S. (2005). A comparison of eigensolvers for large-scale 3D modal analysis using AMG-preconditioned iterative methods. International Journal for Numerical Methods in Engineering *64*, 204-236.

Ashburner, J., and Friston, K. J. (2005). Unified segmentation. NeuroImage *26*, 839-851.

Aubert, G., and Kornprobst, P. (2002). Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations, Vol 147 (New York, Springer-Verlag).

Baker, C. I., Hutchison, T. L., and Kanwisher, N. (2007). Does the fusiform face area contain subregions highly selective for nonfaces? Nat Neurosci *10*, 3-4.

Bamberg, P., and Shlomo, S. (1990). A course in mathematics for students of physics, Vol 2 (Cambridge, Cambridge University Press).

Begelfor, E., and Werman, M. (2005). How to put probabilities on homographies. IEEE Transactions on Pattern Analysis and Machine Intelligence *27*, 1666-1670.

Bibliography

Belkin, M., and Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation *15*, 1373-1396.

Bishop, C. (1995). Neural networks for pattern recognition (Oxford, Oxford University Press).

Bishop, C. (2006). Pattern recognition for machine learning (New York, Springer).

Bishop, C., and Svensen, M. (2003). Bayesian hierarchical mixtures of experts. Paper presented at: Uncertainty in Artificial Intelligence.

Bishop, C. M. (1999). Latent variable models. In Learning in graphical models, M. I. Jordan, ed. (Massachusetts, MIT Press), pp. 371-403.

Brockett, R. (1997). Notes on Stochastic Processes on Manifolds. In Systems and Control in the Twenty-First Century, C. e. a. Byrnes, ed. (Boston, USA, Birkhauser), pp. 75-101.

Buxton, R. B., Wong, E. C., and Frank, L. R. (1998). Dynamics of blood flow and oxygenation changes during brain activation: the balloon model. Magn Reson Med *39*, 855-864.

Canny, J. (1983). Finding Edges and Lines in Images. In Technical Report: AITR-720 (Cambridge, MA, USA, Massachusetts Institute of Technology Cambridge, MA, USA).

Catte, F., Lions, P. L., Morel, J. M., and Coll, T. (1992). Image selective smoothing and edge detection by nonlinear diffusion. SIAM J Numer Anal *29*, 182-193.

Chan, T., and Shen, J. (2005). Image Processing and Analysis: Variational, PDE, Wavelet, and Stochastic Methods (Philadelphia, USA, SIAM).

Chung, F. (1997). Spectral graph theory (Providence, Rhode Island, American mathematics society).

Chung, M. K., Dalton, K. M., Shen, L., Evans, A. C., and Davidson, R. J. (2007). Weighted fourier series representation and its application to quantifying the amount of gray matter. IEEE Trans Med Imaging *26*, 566-581.

Bibliography

Chung, M. K., Worsley, K. J., Robbins, S., Paus, T., Taylor, J., Giedd, J. N., Rapoport, J. L., and Evans, A. C. (2003). Deformation-based surface morphometry applied to gray matter deformation. NeuroImage *18*, 198-213.

Coifman, R. R., and Maggioni, M. (2006). Diffusion wavelets. Applied and Computational Harmonic Analysis *21*, 53-94.

Cornford, D., Csato, L., and Opper, M. (2005). Sequential, Bayesian geostatistics: A principled method for large data sets. Geographical Analysis *37*, 183-199.

Csato, L., and Opper, M. (2002). Sparse on-line Gaussian processes. Neural Computation *14*, 641-668.

Dayan, P., and Abbott, L. (2001). Theoretical neuroscience: Computational and mathematical modeling of neural systems, 1st edn (Cambridge, Massachusetts, The MIT press).

Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. Journal of Royal Statistical Society Series B *39*, 1-38.

Descombes, X., Kruggel, F., and von Cramon, D. Y. (1998). fMRI signal restoration using a spatio-temporal Markov Random Field preserving transitions. Neuroimage *8*, 340-349.

DeYoe, E. A., Bandettini, P., Neitz, J., Miller, D., and Winans, P. (1994). Functional magnetic resonance imaging (FMRI) of the human brain. J Neurosci Methods *54*, 171-187.

Engel, S. A., Rumelhart, D. E., Wandell, B. A., Lee, A. T., Glover, G. H., Chichilnisky, E. J., and Shadlen, M. N. (1994). fMRI of human visual cortex. Nature *369*, 525.

Faugeras, O., Adde, G., Charpiat, G., Chefd'Hotel, C., Clerc, M., Deneux, T., Deriche, R., Hermosillo, G., Keriven, R., Kornprobst, P., *et al.* (2004). Variational, geometric, and statistical methods for modeling brain anatomy and function. Neuroimage *23*, S46-S55.

Flandin, G., Kherif, F., Pennec, X., Malandain, G., Ayache, N., and Poline, J. B. (2002). Improved detection sensitivity in functional MRI data using a brain parcelling technique.

Bibliography

Medical Image Computing and Computer-Assisted Intervention-MICCAI 2002, Pt 1 *2488*, 467-474.

Flandin, G., and Penny, W. D. (2007). Bayesian fMRI data analysis with sparse spatial basis function priors. NeuroImage *34*, 1108-1125.

Friman, O., Borga, M., Lundberg, P., and Knutsson, H. (2003). Adaptive analysis of fMRI data. NeuroImage *19*, 837-845.

Friston, K. (2002). Functional integration and inference in the brain. Prog Neurobiol *68*, 113-143.

Friston, K., Ashburner, J., Frith, C. D., Poline, J. B., Heather, J. D., and Frackowiak, R. S. (1995). Spatial registration and normalization of images. Hum Brain Mapp *3*, 165-189.

Friston, K., Ashburner, J., Kiebel, S., Nichols, T., and Penny, W. (2006). Statistical Parametric Mapping: The analysis of functional brain images (London, Elsevier).

Friston, K., Mattout, J., Trujillo-Barreto, N., Ashburner, J., and Penny, W. (2007). Variational free energy and the Laplace approximation. NeuroImage *34*, 220-234.

Friston, K. J., Fletcher, P., Josephs, O., Holmes, A., Rugg, M. D., and Turner, R. (1998a). Event-related fMRI: characterizing differential responses. Neuroimage *7*, 30-40.

Friston, K. J., Glaser, D. E., Henson, R. N., Kiebel, S., Phillips, C., and Ashburner, J. (2002a). Classical and Bayesian inference in neuroimaging: applications. NeuroImage *16*, 484-512.

Friston, K. J., Josephs, O., Rees, G., and Turner, R. (1998b). Nonlinear event-related responses in fMRI. Magn Reson Med *39*, 41-52.

Friston, K. J., Mechelli, A., Turner, R., and Price, C. J. (2000). Nonlinear responses in fMRI: the Balloon model, Volterra kernels, and other hemodynamics. Neuroimage *12*, 466-477.

Bibliography

Friston, K. J., and Penny, W. (2003). Posterior probability maps and SPMs. NeuroImage *19*, 1240-1249.

Friston, K. J., Penny, W., Phillips, C., Kiebel, S., Hinton, G., and Ashburner, J. (2002b). Classical and Bayesian inference in neuroimaging: theory. NeuroImage *16*, 465-483.

Geman, S., and Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. IEEE-PAMI *6*, 721-741.

Gerig, G., Kubler, O., Kikinis, R., and Jolesz, F. (1992). Nonlinear anisotropic filtering of MRI data. IEEE Transactions on Medical Imaging *11*, 221-232.

Gilbert, J. R., Miller, G. L., and Teng, S. (1998). Geometric mesh partitioning: Implementation and experiments. SIAM Journal of Scientific Computing *19*, 2091-2110.

Goldberg, P. W., Williams, C. K. I., and Bishop, C. (1998). Regression with input-dependent noise: a Gaussian process treatment. Paper presented at: NIPS 10 (MIT Press).

Gossl, C., Auer, D. P., and Fahrmeir, L. (2001). Bayesian spatiotemporal inference in functional magnetic resonance imaging. Biometrics *57*, 554-562.

Grady, L., and Schwartz, E. L. (2003). The graph analysis toolbox: Image processing on arbitrary graphs (Boston, MA, Boston University).

Grady, L., and Schwartz, E. L. (2004). Faster graph-theoretic image processing via small-world and quadtree topologies. Paper presented at: CVPR04 (Washington, DC, IEEE).

Grady, L., and Schwartz, E. L. (2006). Isoperimetric graph partitioning for image segmentation. IEEE Trans Pattern Anal Mach Intell *28*, 469-475.

Grill-Spector, K., Sayres, R., and Ress, D. (2006). High-resolution imaging reveals highly selective nonface clusters in the fusiform face area. Nat Neurosci *9*, 1177-1185.

Gupta, A. K., and Nagar, D. K. (2000). Matrix variate distributions (Boca Raton, Chapman & Hall/CRC).

Bibliography

Harrison, L. M., David, O., and Friston, K. J. (2005). Stochastic models of neuronal dynamics. Philos Trans R Soc Lond B Biol Sci *360*, 1075-1091.

Harrison, L. M., Penny, W., Ashburner, J., Trujillo-Barreto, N., and Friston, K. J. (2007a). Diffusion-based spatial priors for imaging. NeuroImage *38*, 677-695.

Harrison, L. M., Penny, W., Daunizeau, J., and Friston, K. J. (2008). Diffusion-based spatial priors for functional magnetic resonance images. NeuroImage.

Harrison, L. M., Stephan, K. E., Rees, G., and Friston, K. J. (2007b). Extra-classical receptive field effects measured in striate cortex with fMRI. NeuroImage *34*, 1199-1208.

Harville, D. (1997). Matrix algebra from a statistician's perspective (New York, Springer Science+Business Media Inc).

Haynes, J. D., Deichmann, R., and Rees, G. (2005). Eye-specific effects of binocular rivalry in the human lateral geniculate nucleus. Nature *438*, 496-499.

Hendrickson, B., and Leland, R. (1994). The Chaco User's Guide version 2.0.

Henson, R. N., Shallice, T., Gorno-Tempini, M. L., and Dolan, R. J. (2002). Face repetition effects in implicit and explicit memory tests as measured by fMRI. Cereb Cortex *12*, 178-186.

Hollander, I., and Bajla, I. (1998). Adaptive smoothing of MR brain images by 3D geometry-driven diffusion. Computer Methods and Programs in Biomedicine *55*, 157-+.

Jordan, M. I., ed. (1999). Learning in graphical models (Cambridge, Massachusetts, The MIT press).

Karypis, G., and Kumar, V. (1998). A fast and high quality multilevel scheme for partitioning irregular graphs. Siam Journal on Scientific Computing *20*, 359-392.

Kass, R. E., and Raftery, A. E. (1995). Bayes Factors. Journal of the American Statistical Association *90*, 773-795.

Bibliography

Kersting, K., Plagemann, C., Pfaff, P., and Burgard, W. (2007). Most likely heteroscedastic Gaussian process regression. Paper presented at: International Conference on Machine Learning.

Kiebel, S. J., Goebel, R., and Friston, K. J. (2000). Anatomically informed basis functions. NeuroImage *11*, 656-667.

Kiebel, S. J., Poline, J. B., Friston, K. J., Holmes, A. P., and Worsley, K. J. (1999). Robust smoothness estimation in statistical parametric maps using standardized residuals from the general linear model. Neuroimage *10*, 756-766.

Kim, H. Y., and Cho, Z. H. (2002). Robust Anisotropic Diffusion to Produce Clear Statistical Parametric Map from Noisy fMRI. Proceedings of the 15th Brazilian Symposium on Computer Graphics and Image Processing, 11-17.

Kim, H. Y., Javier, G., and Cho, Z. H. (2005). Robust anisotropic diffusion to produce enhanced statistical parametric map from noisy fMRI. Computer Vision and Image Understanding *99*, 435-452.

Kimmel, R. (2003). Numerical geometry of images (New York, Springer).

Knutsson, H. E., Wilson, R., and Granlund, G. H. (1983). Anisotropic Nonstationary Image Estimation and Its Applications.1. Restoration of Noisy Images. IEEE Transactions on Communications *31*, 388-397.

Koenderink, J. J. (1984). The Structure of Images. Biological Cybernetics *50*, 363-370.

Lawrence, N. (2006). Large scale learning with the Gaussian process latent variable model (Technical Report No. CS-06-05. University of Sheffield).

Li, S. Z. (2001). Markov random field modeling in image analysis, 2nd edn (Tokyo, Springer-Verlag).

Lindeberg, T. (1994). Scale-Space Theory in Computer Vision (Stockholm, Sweden, Kluwer Academic Publishers).

Bibliography

Logothetis, N. K., and Wandell, B. A. (2004). Interpreting the BOLD signal. Annu Rev Physiol *66*, 735-769.

MacKay, D. J. C., ed. (1998). Introduction to Gaussian Processes, Neural Networks and Machine Learning edn (Berlin, Springer).

MacKay, D. J. C. (2003). Information theory, inference, and learning algorithms (Cambridge, Cambridge University Press).

Maggioni, M., and Mahadevan, S. (2006). A Multiscale Framework For Markov Decision Processes using Diffusion Wavelets (Massachusetts, University of Massachusetts).

Memoli, F., Sapiro, G., and Thompson, P. (2004). Implicit brain imaging. NeuroImage *23 Suppl 1*, S179-188.

Miezin, F. M., Maccotta, L., Ollinger, J. M., Petersen, S. E., and Buckner, R. L. (2000). Characterizing the hemodynamic response: effects of presentation rate, sampling procedure, and the possibility of ordering brain activity based on relative timing. Neuroimage *11*, 735-759.

Minka, T. (2000). Old and new matrix algebra useful for statistics.

Mobbs, D., Petrovic, P., Marchant, J. L., Hassabis, D., Weiskopf, N., Seymour, B., Dolan, R. J., and Frith, C. D. (2007). When fear is near: threat imminence elicits prefrontal-periaqueductal gray shifts in humans. Science *317*, 1079-1083.

Mohar, B. (1989). Isoperimetric Numbers of Graphs. Journal of Combinatorial Theory Series B *47*, 274-291.

Moler, C., and Van Loan, C. (2003). Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. Siam Review *45*, 3-49.

Mumford, D., and Shah, J. (1989). Optimal Approximations by Piecewise Smooth Functions and Associated Variational-Problems. Communications on Pure and Applied Mathematics *42*, 577-685.

Bibliography

Nair, D. G. (2005). About being BOLD. Brain Res Brain Res Rev *50*, 229-243.

Osher, S., and Paragios, N. (2003). Geometric Level Set Methods in Imaging Vision and Graphics (New York, Springer Verlag).

Patterson, H. D., and Thompson, R. (1974). Maximum likelihood estimation of components of variance. Paper presented at: 8th International Biometrics Conference (Constanta, Romania).

Penny, W., Ashburner, J., Kiebel, S., Henson, R., Glaser, D., Phillips, C., and Friston, K. (2001). Statistical Parametric Mapping: An Annotated Bibliography.

Penny, W., Flandin, G., and Trujillo-Barreto, N. (2007). Bayesian comparison of spatially regularised general linear models. Hum Brain Mapp *28*, 275-293.

Penny, W., and Friston, K. (2003). Mixtures of general linear models for functional neuroimaging. IEEE Trans Med Imaging *22*, 504-514.

Penny, W. D., Trujillo-Barreto, N. J., and Friston, K. J. (2005). Bayesian fMRI time series analysis with spatial priors. NeuroImage *24*, 350-362.

Perona, P., and Malik, J. (1990). Scale-Space and Edge-Detection Using Anisotropic Diffusion. Ieee Transactions on Pattern Analysis and Machine Intelligence *12*, 629-639.

Pollmann, S., Wiggins, C. J., Norris, D. G., von Cramon, D. Y., and Schubert, T. (1998). Use of short intertrial intervals in single-trial experiments: a 3T fMRI-study. Neuroimage *8*, 327-339.

Polzehl, J., and Spokoiny, V. G. (2001). Functional and dynamic magnetic resonance imaging using vector adaptive weights smoothing. Journal of the Royal Statistical Society Series C-Applied Statistics *50*, 485-501.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (2007). Numerical Recipes, 3rd edn (Cambridge, U.K, Cambridge University Press).

Bibliography

Prewitt, J. M. S. (1970). Object enhancement and extraction. In Picture Processing and Psychopictories, B. S. Lipkin, and A. Rosenfeld, eds. (New York, Academic Press).

Qiu, A., Bitouk, D., and Miller, M. I. (2006). Smooth functional and structural maps on the neocortex via orthonormal bases of the Laplace-Beltrami operator. IEEE Trans Med Imaging *25*, 1296-1306.

Qui, H., and Hancock, E. R. (2007). Clustering and Embedding Using Commute Times. IEEE Transactions on Pattern Analysis and Machine Intelligence *29*, 1873-1890.

Quinonero-Candela, J. Q., and Rasmussen, C. E. (2005). A unifying view of sparse approximate Gaussian process regression. Journal of Machine Learning Research *6*, 1939-1959.

Rasmussen, C., and Williams, C. (2006). Gaussian processes for machine learning (Cambridge, Massachusetts, The MIT Press).

Roberts, L. (1965). Machine perception of three-dimensional solids. In Optical and Electro-optical Information Processing, J. Tippett, ed. (Cambridge, MIT Press), pp. 157--197.

Romeny, B. M. T. (1994). Geometry-driven diffusion in computer vision (Dordrecht, The Netherlands, Kluwer Academic Publishers).

Romeny, B. M. T. (2003). Front-End Vision & Multi-Scale Image Analysis (Dordrecht, The Netherlands, Kluwer Academic Publishers).

Rosenberg, S. (1997). The Laplacian on a Riemannian Manifold (Cambridge, U.K, Cambridge University Press).

Rossmann, W. (2002). Lie Groups: An Introduction through Linear Groups (Oxford, Oxford University Press).

Rudin, L. I., Osher, S., and Fatemi, E. (1992). Nonlinear Total Variation Based Noise Removal Algorithms. Physica D *60*, 259-268.

Bibliography

Scherzer, O., Grasmair, M., Grossauer, H., Haltmeier, M., and Lenzen, F. (2008). Variational Methods in Imaging, Vol 650 (New York, Springer-Verlag).

Schneider, K. A., and Kastner, S. (2005). Visual responses of the human superior colliculus: a high-resolution functional magnetic resonance imaging study. J Neurophysiol *94*, 2491-2503.

Sereno, M. I., McDonald, C. T., and Allman, J. M. (1994). Analysis of retinotopic maps in extrastriate cortex. Cereb Cortex *4*, 601-620.

Shafie, K., Sigal, B., Siegmund, D., and Worsley, K. J. (2003). Rotation space random fields with an application to fMRI data. Annals of Statistics *31*, 1732-1771.

Shi, J. B., and Malik, J. (2000). Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence *22*, 888-905.

Simmons, W. K., Bellgowan, P. S., and Martin, A. (2007). Measuring selectivity in fMRI data. Nat Neurosci *10*, 4-5.

Smith, S. M., and Brady, J. M. (1997). SUSAN - A new approach to low level image processing. International Journal of Computer Vision *23*, 45-78.

Smith, S. T. (2005). Covariance, subspace, and intrinsic Cramer-Rao bounds. IEEE Transactions on Signal Processing *53*, 1610-1630.

Snelson, E., and Ghahramani, Z. (2006). Sparse Gaussian processes using pseudo-inputs. Paper presented at: Neural Information Processing Systems (NIPS) (MIT Press).

Snelson, E., and Ghahramani, Z. (2007). Local and global sparse Gaussian process approximations. Artificial Intelligence and Statistics (AISTATS) *11*.

Snelson, E., Rasmussen, C. E., and Ghahramani, Z. (2003). Warped Gaussian processes (Gatsby Computational Neuroscience Unit).

Bibliography

Sobel, I., and Feldman, G. (1973). A 3x3 Isotropic Gradient Operator for Image Processing. In Pattern Classification and Scene Analysis, R. Duda, and P. Hart, eds. (John Wiley and Sons), pp. 271-272.

Sochen, N., Kimmel, R., and Malladi, R. (1997). From high energy physics to low level vision. Scale-Space Theory in Computer Vision *1252*, 236-247.

Sochen, N., Kimmel, R., and Malladi, R. (1998). A general framework for low level vision. IEEE Transactions on Image Processing *7*, 310-318.

Sole, A. F., Ngan, S. C., Sapiro, G., Hu, X., and Lopez, A. (2001). Anisotropic 2-D and 3-D averaging of fMRI signals. IEEE Trans Med Imaging *20*, 86-93.

Strang, G. (2004). Linear Algebra and Its Applications (Belmont, USA, Thomson Brookes/Cole).

Strang, G. (2007). Computational Science and Engineering, Wellesley-Cambridge Press).

Stuben, K. (2001). A review of algebraic multigrid. Journal of Computational and Applied Mathematics *128*, 281-309.

Sylvester, R., Josephs, O., Driver, J., and Rees, G. (2007). Visual FMRI responses in human superior colliculus show a temporal-nasal asymmetry that is absent in lateral geniculate and visual cortex. J Neurophysiol *97*, 1495-1502.

Tabelow, K., Polzehl, J., Voss, H. U., and Spokoiny, V. (2006). Analyzing fMRI experiments with structural adaptive smoothing procedures. NeuroImage *33*, 55-62.

Talairach, J., and Tournoux, P. (1988). Co-planar Stereotaxic Atlas of the Human Brain: 3-Dimensional Proportional System - an Approach to Cerebral Imaging (New York, Thieme Medical Publishers).

Taylor, J., Worsley, K. J., Chung, M., and Evans, A. C. (2001). Thresholding non-stationary SPMs with an application to cortical surface mapping. NeuroImage *13*, S264-S264.

Bibliography

Taylor, J. E., and Worsley, K. J. (2007). Detecting sparse signals in random fields, with an application to brain mapping. Journal of the American Statistical Association *102*, 913-928.

Tenenbaum, J. B., de Silva, V., and Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. Science *290*, 2319.

Teo, P. C., Sapiro, G., and Wandell, B. A. (1997). Creating connected representations of cortical gray matter for functional MRI visualization. IEEE Trans Med Imaging *16*, 852-863.

Thirion, B., Flandin, G., Pinel, P., Roche, A., Ciuciu, P., and Poline, J. B. (2006). Dealing with the shortcomings of spatial normalization: multi-subject parcellation of fMRI datasets. Hum Brain Mapp *27*, 678-693.

Tipping, M. E. (2004). Bayesian inference: An introduction to principles and practice in machine learning. Advanced Lectures on Machine Learning *3176*, 41-62.

Trujillo-Barreto, N. J., Aubert-Vazquez, E., and Valdes-Sosa, P. A. (2004). Bayesian model averaging in EEG/MEG imaging. NeuroImage *21*, 1300-1319.

Ueda, N., and Ghahramani, Z. (2002). Bayesian model search for mixture models based on optimizing variational bounds. Neural Networks *15*, 1223.

Walker, S. A., Miller, D., and Tanabe, J. (2006). Bilateral spatial filtering: refining methods for localizing brain activation in the presence of parenchymal abnormalities. NeuroImage *33*, 564-569.

Wand, M. P. (2002). Vector differential calculus in statistics. American Statistician *56*, 55-62.

Warnking, J., Dojat, M., Guerin-Dugue, A., Delon-Martin, C., Olympieff, S., Richard, N., Chehikian, A., and Segebarth, C. (2002). fMRI retinotopic mapping--step by step. NeuroImage *17*, 1665-1683.

Weickert, J. (1998). Anisotropic diffusion in image processing (Stuttgart, Teubner-Verlag).

Bibliography

Witkin, A., and Witkin, A. (1984). Scale-space filtering: A new approach to multi-scale description
Scale-space filtering: A new approach to multi-scale description. Paper presented at: Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '84.

Woolrich, M. W., Jenkinson, M., Brady, J. M., and Smith, S. M. (2004). Fully Bayesian spatio-temporal modeling of FMRI data. IEEE Trans Med Imaging *23*, 213-231.

Worsley, K. J., Andermann, M., Koulis, T., MacDonald, D., and Evans, A. C. (1999). Detecting changes in nonisotropic images. Hum Brain Mapp *8*, 98-101.

Worsley, K. J., Marrett, S., Neelin, P., and Evans, A. C. (1996a). Searching scale space for activation in PET images. Human Brain Mapping *4*, 74-90.

Worsley, K. J., Marrett, S., Neelin, P., Vandal, A. C., Friston, K. J., and Evans, A. C. (1996b). A unified statistical approach for determining significant voxels in images of cerebral activation. Hum Brain Mapp *4*, 58-73.

Zhang, F., and Hancock, E. R. (2005). Image scale-space from the heat kernel. Progress in Pattern Recognition, Image Analysis and Applications, Proceedings *3773*, 181-192.

Zhang, F., and Hancock, E. R. (2006). Riemannian graph diffusion for DT-MRI regularization. Medical Image Computing and Computer-Assisted Intervention - MICCAI 2006, Pt 2 *4191*, 234-242.

Zhang, F., and Hancock, E. R. (2007). Graph Spectral Image Smoothing. In Graph-Based Representations in Pattern Recognition, F. Escolano, and M. Vento, eds. (Berlin / Heidelberg, Springer-Verlag), pp. 191-203.

Zhu, S. C., and Mumford, D. (1997). Prior learning and gibbs reaction-diffusion. IEEE Trans PAMI *19*, 1236-- 1250.